Some Statistical Techniques for Digital Image Analysis

THESIS

Submitted For the Award of the Degree Of

Doctor of philosophy

In

Statistics

Fatima Siddiqui

By

Under The Supervision Of

Prof. Qazi Mazhar Ali

DEPARTMENT OF STATISTICS
ALIGARH MUSLIM UNIVERSITY
ALIGARH (INDIA)
2016

Acknowledgements

First and foremost, I praise and thank ***Allah***, *The Almighty*, for all the blessings He has bestowed upon me. Next I want to place on record my profound gratitude and heartfelt thanks to my supervisor *Prof. Qazi Mazhar Ali*, Chairperson, Department of Statistics, Aligarh Muslim University for his valuable guidance, encouragement, and for providing all the necessary facilities throughout my research period. It has been an honor to be his Ph.D. student. I appreciate and thank him for encouraging my research and for allowing me to grow as a researcher. I will always be grateful for his critical but valuable advice. I thank him for understanding me and helping me through highs and lows of the research period. His unflinching patience and conviction has always inspired me. I thank him for his constructive criticism, extensive discussions and advice regarding my research work.

I would also like to express my gratitude to all the other faculty members of the department, *Prof. Mohammad Jameel Ahsan, Prof. Abdul Bari, Prof. Athar Ali Khan, Prof. Ariful Islam, Prof. Aqueel Ahmad, Mr. Syed Suhaib Hasan, Dr. Rafiqullah Khan, Dr. Haseeb Athar, Dr. Shakeel Javaid, Dr. Irfan Khan, Dr. Md. Jahangir S. Khan, Dr. Ahmad Yusuf Adhami, Dr. Md. Arshad,. Dr. Zaki Anwar* and *Dr. Mohammad Faizan* for rendering their whole-hearted cooperation and help whenever needed throughout this journey of mine. I specially thank *Prof. Athar Ali Khan* for devoting a great deal of his time while helping me with R software.

I extend my heartfelt gratitude to *Dr. Omar Farooq* and *Dr. Tauqueer Ahmad* for extending their time, support and cooperation for the sake of successful completion of my research.

I owe a great deal of appreciation to all the non-teaching staff members of the Department of Statistics & O.R., AMU, for the help and cooperation they rendered during my research period.

I would not have been able to reach this point in my life, had I not been supported by my parents *Mr. Mohd. Ammar Siddiqui* and *Mrs. Shagufta*

Ammar. They compromised with the most beautiful period of their life for the sake of my progress. Mere words cannot express how grateful I am to them. I wholeheartedly pray ***Allah**, The Almighty* to give me guidance so that I could reciprocate the same for them.

I cannot thank my brothers, *Ayaz, Faraz, Saim* and my sisters, *Ms. Ayesha* and *Wajiha* enough. Though separated by distances, they have always motivated me through everything in the very specific "siblings-way. My loving and very supportive cousins *Abdullah* and *Zaid* deserve my special thanks as they helped me with my work at a real crucial time.

I owe special thanks to my friend, *Mr. Abdul Samad* for believing in me and for being the very first person who motivated me for pursuing research. I thank him for reviving my confidence levels whenever I fell short on it through the most stressful periods of this journey and for being always there for me, no matter what, like a true friend.

Its my pleasure to specially thank *Sumeera, Romana, Shaista* and *Sana* for being friends in need. My roommates *Ms. Sheeba* and *Shehroora* deserve a special mention who were very supportive and understanding as roommates. I am grateful to all of them for playing the stress busters in my life, for their motivational words and for everything they did for me that made me enjoy my research period.

I am grateful to all the senior research scholars, my colleagues, as well as my juniors especially *Dr. Sanam Haseen, Dr. Sana Iftekhar, Dr. Farah Naz, Dr. Zubdah-e-Noor, Dr. Abdul Quddoos* and *Mrs. Sheema Sadia* for their encouragements, discussions and fruitful suggestions. I am specially thankful to my junior *Mr Ateeb-ur-rehman Sherwani* who devoted his time for initial proof reading of this document.

I take this opportunity to sincerely acknowledge the *University Grants Commission*, New Delhi for the financial assistance they provided in the form of *Maulana Azad National Fellowship*, for completing this research work.

I whole heartedly thank all my well wishers for all the love and prayers they showered on me. I thank all the people who have helped me directly or indirectly in this research endeavor.

Aligarh, May 2016 Fatima Siddiqui

Preface

A ***digital image*** is an image that has been discretized both in spatial coordinates as well as in brightness values and can be manipulated and evaluated electronically with computers. It is not very long enough when digitizing an image and saving it to a computer was a time consuming task. But, the availability of powerful computers on every desktop and the modern programming environments make practically every aspect of computing in digital imaging, easily available to non-expert users. All of these developments have resulted in a large community of researchers that works productively with digital images while having only a basic knowledge of the underlying concepts and mechanics.

With applications like optical character recognition based automated billing in shops, image analysis technology has become a reliable and indispensable element in our daily lives. Digital image analysis techniques aim at extracting meaningful information about the contents of an appropriately pre-processed digital image.

A typical digital image analysis task requires the extraction of certain features (which can be spectral, spatial, textural or shape features) that aid in identification of the objects in the imaged scene. Although, the task of digital image analysis basically involves the study of *feature extraction, segmentation*, and *classification* techniques, this thesis discusses and emphasizes upon some of the various statistical as well as non-statistical, classification techniques developed in the field of image processing so far.

Image classification research focussing on developing improved methods for classification of images has long attracted the attention of researchers because classification results are the basis for many environmental and socio-economic applications. And hence, scientists and researchers have given in a great deal of efforts in developing advanced classification methodologies as well as in improving the existing ones. The aim of this thesis is to present a thorough review of various classical and advaned machine learning classification techniques used for classification of digital images and to measure the effect of

skewness on these classification methods .

Chapter 1 gives an overview of the various terminologies and basic underlying methodologies in a digital image analysis task. The foci of this chapter are on discussing and providing a summary of some of the major statistical as well as non-statistical methods and techniques used for image classification, methods used for assessing and improving classification accuracies, and on discussing the issues affecting the success of these techniques.

Chapter 2 is based on the article entitled, "Performance of Non-Parametric Classifiers on Highly Skewed Data", published in *Global Journal of Pure and Applied Mathematics* (2016), Volume **12**, Issue **3**, pages $1547 - 1565$. It discusses some significant studies which highlight the limitations of the most widely used parametric Maximum likelihood classifier (MLC). The multivariate normal distributional assumptions of the MLC are often found to be violated in real life situations. Instead, the real life datasets are often found to be skewed in nature. In such non-optimal situations for the parametric MLC, the analysts often look out for the non-parametric alternatives. This chapter discusses in detail some of the most advanced non-parametric classifiers i.e. artificial neural networks (ANNs), support vector machines (SVMs) and random forest (RF) classifiers in the field of image analysis. The major advantage of these classifiers over MLC is that they neither assume any statistical probability distribution for the data classes nor require any statistical parameter estimation to separate the classes and hence guarantee better classification outcomes (Paola and Schowengerdt, 1995; Foody, 2002), when the underlying data is not normal or specifically skewed in the context of this thesis. Further, the investigations based on extensively simulated datasets as well as some real datasets were carried out in this chapter to evaluate and compare the performances of these classifiers while classifying highly skewed datasets.

Chapter 3 is based on the article entitled, "An Analytical Study of the Classification of Highly Skewed Data" which is under review in *Communications in Statistics: Simulation and Computation*. This chapter identifies some of the grave concerns that limit the wide scale adaptability of the advanced non-parametric classifiers ANN, SVM and RF, discussed in Chapter 2. This chapter further highlights the strengths of the MLC which make it the most popular classification technique among the practitioners. A thorough in depth review of the literature of classification methodologies used in image analysis has been discussed in this chapter. This review notices the lack of any study measuring the effect of severe skewness of the underlying data classes on the

classification accuracies of the MLC. Hence, this chapter tries to fill this gap by thoroughly investigating the effects of various data characteristics along with skewness of data classes on the performance of MLC via simulated as well as real datasets. Acknowledging the caliber of the lognormal distributions of describing many forms of experimental data which are asymmetrically distributed (Aitchison and Brown, 1963; Gale, 1967; Crow and Shimizu, 1988), this chapter proposes a new discriminant function based on the multivariate lognormal distribution for efficient classification of severely skewed datasets. A simple computer aided algorithm which enables the testing of underlying data classes in a dataset for assessing normality assumption of MLC and accordingly decides to use the conventional linear discriminant function, quadratic discriminant function or the suggested discriminant function has also been suggested in the chapter.

Chapter 4 is based on the research article entitled, "A New Transformation for Normalizing Skewed Data in Classification Problems Based on the Multivariate 3-Parameter Lognormal Distribution", communicated to *Journal of Classification*, Springer. It highlights the importance of employing data transformations in improving the performances of parametric as well as non-parametric classifiers. This chapter further discusses some of the most popular data transformations developed in the literature and their limitations. Regular lognormal transformations based on 2-parameter lognormal distribution are most frequently used for restoring normality in skewed datasets. But, these transformations are incapable of handling negatively skewed as well negative valued and zero valued observations. This chapter proposes a new set of data transformations based on the multivariate 3-parameter lognormal distribution and illustrate their efficiency in transforming negatively as well as positively skewed datasets spread over the whole of real line via simulated as well as real image datasets.

Aligarh, May 2016 Fatima Siddiqui

Contents

Acknowledgements **iii**

Preface **vii**

Contents **xi**

List of Figures **xv**

List of Tables **xix**

1 An Overview of the Basic Concepts of Digital Image Analysis **1**

1.1 Introduction 1
1.1.1 Applications of digital image processing 2
1.1.2 Digital image processing 3
1.1.3 Digital image analysis 4
1.1.4 Interpretation of digital images 6
1.2 Classification 6
1.2.1 Unsupervised classification techniques 7
1.2.2 Supervised classification techniques 11
1.2.3 Parametric classification techniques 12
1.2.4 Non-parametric classification techniques 14
1.2.5 Hybrid unsupervised/supervised classification approach . 16
1.2.6 Advanced supervised classification methods 17
1.2.7 Ensemble methods for classification 20
1.2.8 Classification of hyper-dimensional imagery 21
1.2.9 Feature reduction 21
1.3 Assessment of Classification Accuracy 23
1.3.1 Error matrix 23
1.3.2 Misclassification error rates 25

1.3.3 Agreement measures 27
1.3.4 Area under ROC curve (AUROC) 28
1.4 Data Considerations for Supervised Statistical Classifiers 31
1.4.1 Sampling scheme 31
1.4.2 Sample size 32
1.4.3 Adequacy of training data 33

2 Performance of Non-Parametric Classifiers on highly skewed data 35
2.1 Introduction 35
2.1.1 Motivation 36
2.1.2 Objective of the study 38
2.1.3 Non-parametric alternatives to parametric classifiers . . . 38
2.2 Background 40
2.3 Non-Parametric Classifiers and Other Methods Used 42
2.3.1 Artificial neural networks (ANNs) 42
2.3.2 Support vector machines (SVMs) 45
2.3.3 Random forests (RFs) 52
2.3.4 Accuracy assessment 56
2.4 Numerical Experiments and Results 57
2.4.1 Simulation and data generation 57
2.4.2 Real datasets used for comparison 59
2.4.3 Results 62
2.5 Conclusion 66

3 New Discrminant Function and Methodology for Classification of Highly Skewed Data 75
3.1 Introduction 75
3.1.1 Limitations of non-parametric classifiers 76
3.1.2 Motivation, objective and scope of the study 77
3.2 Background 79
3.3 Maximum Likelihood Classifier and the Suggested Methodology 80
3.3.1 Maximum likelihood classifier 80
3.3.2 Suggested methodology 84
3.3.3 Mardia's test 86
3.3.4 Accuracy assessment 87
3.4 Numerical Illustration and Results 89

3.4.1 Simulation and data generation 89
3.4.2 Real datasets used 90
3.4.3 Results 90
3.5 Conclusion 94

4 A New Transformation for Normalizing Skewed Data in Classification Problems 103
4.1 Introduction 103
4.1.1 Motivation 104
4.2 Background and Scope of the Study 104
4.3 3-Parameter Lognormal Distribution and Suggested Transformations 107
4.3.1 3-parameter lognormal distribution 107
4.3.2 Negatively skewed lognormal distribution 108
4.3.3 Maximum likelihood estimation of $\boldsymbol{\tau}, \boldsymbol{\theta}, \boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ 109
4.3.4 Suggested transformations 111
4.4 Numerical Illustration 112
4.4.1 Data generation via simulation 112
4.4.2 Results 114
4.4.3 Application 116
4.5 Conclusion 116

Bibliography 127

List of Figures

1.1 Various steps involved in the processing/analysis of a digital image. 5

1.2 Types of classifiers. 8

1.3 Unsupervised classification of a satellite image (Adapted from Rais (2015)). 9

1.4 k-means clustering with Mahalanobis distance measures. 10

1.5 Supervised classification of a satellite image (Adapted from Rais (2015)). 12

1.6 Parallelepipeds formation with the issue of overlapping parallelepipeds. 15

1.7 Working of minimum distance classifier with Euclidean distances. As shown, the pixel is at minimum distance from class 1 and hence is labelled as belonging to it by the minimum distance classification algorithm. 16

1.8 An $(m \times m)$ confusion matrix. 24

1.9 ROC plot for four classifiers labelled as A, B, C and D. 30

2.1 A three layer multilayer perceptron network and the typical working of a processing node in forward propagation. 44

2.2 Linear separating hyperplanes for completely separable classes. . 47

2.3 Linear separating hyperplanes for partially separable classes using soft margin concept 49

2.4 Separating hyperplane for inseparable classes using higher dimensional feture space. 50

2.5 A simple decision tree for the classification of iris data. 53

2.6 Working of a random forest. 55

2.7 Plots of expected actual error rates of SVM, RF and ANN over simulated index sample for $\delta = .5$ depicting the effect of training sample size on error rates 68

2.8 Plots of expected actual error rates of SVM, RF and ANN over simulated index sample for $n = 25$, $p = (2, 10)$ and $\delta = .5$ depicting the effect of variability on error rates. 69

2.9 Plots of expected actual error rates of SVM, RF and ANN over simulated index sample for $n = 100$, $p = (2, 10)$ and $\delta = .5$ depicting the effect of data skewness on error rates. 70

3.1 Suggested automated classification mechanism. 86

3.2 Plots of expected actual error rates of LDF,QDF,LNDF,SVM,RF and ANN over simulated index sample for $n = 25$, $p = (2, 10)$ and $\delta = .5$ depicting the effect of data skewness on error rates. . 96

3.3 Plots of expected actual error rates of LDF,QDF,LNDF,SVM,RF and ANN over simulated index sample for $n = 100$, $p = (2, 10)$ and $\delta = .5$ depicting the effect of data skewness on error rates. . 97

3.4 Plots of expected actual error rates of SVM, RF and ANN over simulated index sample for $\delta = .5$ depicting the effect of separability on error rates 98

4.1 Plots of expected actual accuracies of LDF, QDF, QDF on log-transformed data (LogTr) and QDF on 3−parameter log-transformed data over positively skewed simulated index sample for $n = 25$, and $\mu_2 = (\mu_{21}, \mu_{23}$ and $\mu_{25})$ depicting the effect of variability on accuracies. 119

4.2 Plots of expected actual accuracies of LDF, QDF, QDF on log-transformed data (LogTr) and QDF on 3−parameter log-transformed data over positively skewed simulated index sample for $n = 50$, and $\mu_2 = (\mu_{21}, \mu_{23}$ and $\mu_{25})$ depicting the effect of variability on accuracies. 120

4.3 Plots of expected actual accuracies of LDF, QDF, QDF on log-transformed data (LogTr) and QDF on 3−parameter log-transformed data over negatively skewed simulated index sample for $n = 25$, and $\mu_2 = (\mu_{21}, \mu_{23}$ and $\mu_{25})$ depicting the effect of variability on performance of LDF and QDF on transformed and untransformed data. 121

4.4 Plots of expected actual accuracies of LDF, QDF, QDF on log-transformed data (LogTr) and QDF on 3−parameter log-transformed data over negatively skewed simulated index sample for $n = 50$, and $\mu_2 = (\mu_{21}, \mu_{23}$ and $\mu_{25})$ depicting the effect of variability on performance of LDF and QDF on transformed and untransformed data. 122

List of Tables

1.1 Accuracy measures calculated from a (2×2) error matrix. The two classes being referred to as positive class and the negative class. Here a, b, c, d, N are the components $a_{11}, a_{12}, a_{21}, a_{22}, N$ respectively of the error matrix as described in Figure 1.8. . . . 29

2.1 Parameter combinations for simulations. 59

2.2 Apparaent error rate (APER) and Actual error rate (AER) of ANN, SVM and RF with their respective training parameter values for the real datasets. 66

2.3 Classwise Mardia's multivariate coefficient of skewness (S_k) for simulated bivariate index samples 67

2.4 Classwise Mardia's multivariate coefficient of skewness (S_k) for simulated ten variate index samples 67

2.5 Misclassification error rates (in %) of SVM, RF and ANN for simulated skewed data for $(p = 2, \delta = 0.5)$ 71

2.6 Misclassification error rates (in %) of SVM, RF and ANN for simulated skewed data for $(p = 2, \delta = 0.9)$ 72

2.7 Average AC1 statistic of ANN, SVM and RF for simulated skewed data with $(p = 2, \delta = .5)$ 73

3.1 Apparaent error rate, Actual error rate of LDF, QDF and proposed LNDF based classifier for the real datasets. 93

3.2 Expected Apparent and Actual error rates (in %) of LDF, QDF and LNDF for simulated skewed data for $p = 2, \delta = .5$ under Skewed vs Skewed population setting. 99

3.3 Expected Apparent and Actual error rates (in %) of LDF, QDF and LNDF for simulated skewed data for $p = 10, \delta = .5$ under Skewed vs Skewed population setting. 100

3.4 Expected Apparent and Actual error rates (in %) of LDF, QDF and LNDF for simulated skewed data for $p = 2, \delta = .5$ under Normal vs Skewed population setting. 101
3.5 Average AC1 statistic of LDF, QDF and LNDF for simulated skewed data with $(p = 2, \delta = .5)$ 102

4.1 Values of data characteristics used for simulating second population. 114
4.2 Estimated threshold parameters for the significantly skewed training datasets of SPOT data. 117
4.3 Mardia's multivariate coefficient of skewness for second population of simulated positively and negatively skewed datasets. . . . 118
4.4 Expected Apparent Accuracies (APAcs) and Expected Actual Accuracies (AAcs) (in %) of LDF, QDF, LogTr and 3LogTr for simulated positively skewed data. 123
4.5 Expected Apparent Accuracies (APAcs) and Expected Actual Accuracies (AAcs) (in %) of LDF, QDF, LogTr and 3LogTr for simulated positively skewed data. 124
4.6 Area under ROC curve and Gwet's AC1 statistic values of LDF, QDF, LogTr and 3LogTr classifiers for simulated positively skewed data 125
4.7 Area under ROC curve and Gwet's AC1 statistic values of LDF, QDF, LogTr and 3LogTr classifiers for simulated negatively skewed data. 126

1

An Overview of the Basic Concepts of Digital Image Analysis

1.1 Introduction

Digital image processing is a fast developing cross disciplinary research area with growing applications in science and engineering owing to the fact that personal computers and workstations have become powerful enough to process digital image data and less expensive at the same time, so that widespread applications for digital image processing can emerge. Digital image processing has expanded and is further rapidly expanding from a few specialized applications like, astronomy, photogrammetry and particle physics, into a standard scientific tool for analyzing image data in all areas of natural sciences. Before continuing with explaining the various steps involved in a digital image processing task, we briefly describe a digital image.

A ***digital image*** is an image that has been discretized both in spatial coordinates as well as in brightness. Mathematically, a digital image is represented as a two dimensional integer array $f(x, y)$ or a series of such arrays, one for each spectral band in case of multispectral images. Where, x and y are the spatial coordinates of a point in the image and value of f at (x, y) is proportional to the brightness of the image or scene at that point. Each element of the array $f(x, y)$ is called as *pixel* or *picture element* and the digitized brightness values are called the *grey levels*. Thus, a single band digital image

of size $N \times N$ is reptresented as

$$f(x,y) = \begin{bmatrix} f(1,1) & \dots & f(1,N) \\ \vdots & \ddots & \vdots \\ f(N,1) & \dots & f(N,1) \end{bmatrix} \tag{1.1}$$

with,

$$0 \leq f\,(x,y) \leq G-1 \tag{1.2}$$

where G is an integer usually denoting the total number of grey levels in the image.

1.1.1 Applications of digital image processing

There is a plethora of scientific and technical applications of digital image processing, such as remote sensing via satellites and other spacecrafts, image transmission and storage for business applications, medical processing and diagnosis, microscopic imaging, oceanography, sonar and acoustic image processing, robotics, pattern recognition, and automated inspection of industrial parts. Some of these applications discussed here prove that image processing techniques enable investigations of some complex phenomena, which the conventional measuring techniques are not capable of accomplishing.

- Analysis of satellite acquired images or the remotely sensed images are used in tracking of earth resources, geographical mapping, prediction of agricultural crops, urban growth, and weather, flood and fire control, estimating damage in an area due to some natural disaster and also for other environmental applications.

- Digital processing of space images aid in recognition and analysis of objects contained in images obtained from deep space-probe missions. Radar and sonar images aid in guiding aircrafts or missile systems and in detection and recognition of various targets.

- In medical field, analysis of images such as X-rays, angiograms, high resolution digitized mammograms, images of transaxial tomography and other nuclear magnetic resonance (NMR), ultrasonic scanning and radiology images essentially helps in screening, monitoring and in detection of tumors and other diseases in patients.

- Biometric-based identification and verification systems have become a key technology, with applications including controlling access to buildings and computers, reducing fraudulent transactions in electronic commerce, and discouraging illegal immigration. In biometrics, image processing techniques such as image understanding and pattern recognition are required for identifying an individual whose biometric signature is stored in the database previously as an image. Faces, fingerprints, irises, etc., are some of the most actively used image-based biometrics.

- Transmission and storage of digital image data through image processing techniques find their applications in television broadcasts, closed-circuit television based security monitoring systems, transmission of facsimile images for office automation, communication over computer networks, teleconferencing, and in military communications.

- Apart from them, in the field of character recognition, image processing techniques are used for text recognition, mail sorting, label reading, supermarket-product billing etc. In industry, digitized images are used for fault detection and parts identification on assembly lines. In forensics, image processing finds application for finger-print matching and analysis of automated security systems.

1.1.2 Digital image processing

Digital image processing is defined as the conception, design, and enhancement of digital imaging techniques and their practical implementation through computer aided programs. Image processing is not a one-step process, it rather involves several hierarchical steps which are performed in succession until the required data or results are extracted from the observed image. Thus, the whole of image processing tasks can be summarized in a hierarchical system as shown in Figure 1.1. As depicted in the figure, the very first step of image processing is of *image acquisition* where the aim is to capture an image with a suitable, not necessarily optical, acuistion system. Once the image is sensed, it is then digitized so that it can be brought into a form known to computers, this process is called *digitization* of an image. This digitized image is then stored as an array of binary digits in computer memory and is called as a *digital image*. In the successive steps, this digital image is operated upon with some *image preprocessing* tools such as image transformations, noise filters etc.

for regularization, restoration of geometrical distortions and radiometric and geometric calibration of these images. At the next step of *image enhancement* these geometrically corrected and restored image are then enhanced through contrast and edge enhancements, pseudocoloring, sharpening and magnifying for subsequent analysis. And the last step is that of *image analysis*, at which quantitative measurements are obtained from an image through *feature extraction*, *segmentation* and *classification* techniques in order to provide useful descriptions of the image.

1.1.3 Digital image analysis

Even though the digital image analysis is often used interchangeably with digital image processing, from the above description, literally *digital image analysis* be thought of as an essential integral step among the several others in the field of digital image processing as shown in Figure 1.1. Digital image anlaysis techniques aim at extracting meaningful information about the contents of an appropriately preprocessed digital image. In simplest situations, these informations could be distinguishing an object from its background, following a street on a map, finding the bar code on a product, sorting different parts on an assembly line or measuring the size and orientation of blood cells in a medical image (Jain, 1989). While in more sophisticated vision systems, the quantitative measurements obtained through image analysis can be used to take more sophisticated decisions like, controlling the arm of a robot, or navigating an aircraft with the aid of images along its trajectory. Infact, the ultimate aim of most of the image processing applications discussed in Section 1.1.1 in the field of medical image analysis, character recognition, industrial automation, forensics, cartography and remote sensing is to extract features from image data, in order to describe, interpret and understand the various objects and their relationships in the imaged scene in a more knowledgeable way. Thus, *digital image analysis* can be regarded as the most informative part of an image processing task.

A typical digital image analysis task requires the extraction of certain features (which can be spectral, spatial, textural or shape features) that aid in identification of the objects in the imaged scene. These features are employed by segmentation techniques for segmenting the whole image into its components so that quantitative measurements can be obtained on each of the components. And, hence segmented image is then provided to the appropriate

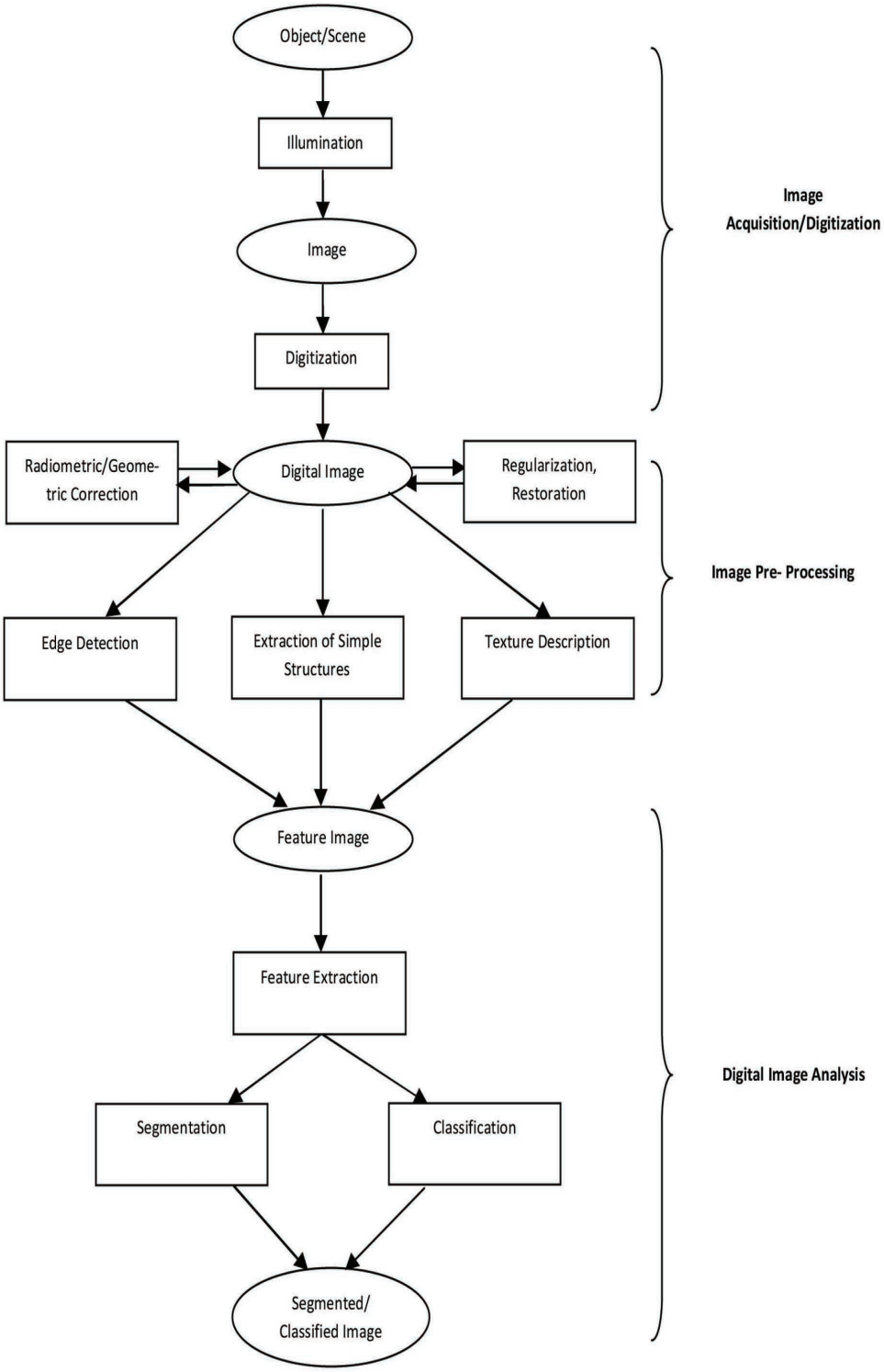


Figure 1.1: Various steps involved in the processing/analysis of a digital image.

classifier which maps different regions of the image into one of the several objects.

Although, the task of digital image analysis basically involves the study of *feature extraction, segmentation*, and *classification* techniques, this thesis discusses and emphasizes upon only the various statistical as well as non-statistical, classification techniques developed in the field of image processing so far. This chapter encompasses the fundamental concepts of a classification problem in digital image analysis and provides an extensive review and introduction of the classifiers that have been produced over the years.

1.1.4 Interpretation of digital images

When the image data is available in digital form, spatially and radiometrically quantized into pixels and brightness levels respectively, it needs to be interpreted for extracting useful information such as estimate of the area under a crop in an image. Two main approaches used for interpreting digital imagery are *photointerpretation* and *quantitative interpretation*. Photointerpretation approach relies on a human analyst or interpreter for extraction of information through visual inspection. The success of this approach depends upon the expertise of the analyst in effectively exploiting the spatial, spectral and temporal information present in the image. Owing to the inability of a human interpreter to discriminate the limit of radiometric resolution available in high resolution images and to process large amount of digital image data as in land cover satellite images, at pixel level, photointerpretation can be effective only for global assessment of geometric characteristics and general appraisal of an image. On the other hand, the quantitative approach, generally referred to as *classification* in digital image analysis field, utilizes the high computational abilities of a computer for identifying each individual pixel in an image with respect to its full radiometric resolution and multidimensional aspect. Among the various frameworks used for formulating a classification problem, the statistical approach is the most extensively studied and the most widely used one.

1.2 Classification

A digital image may contain a number of spectral or information classes. The aim of classification is to identify separable classes in an image, create deci-

sion boundaries between the classes and to establish a relationship between a vector of features describing a pixel or a group of pixels in the image and a class label. The features describing the pixel or a group of pixels may be spectral reflectance, textural measurements derived from the image or geographical features such as elevation, terrain slope etc. Thus, a typical per-pixel classification task in the field of digital image anlaysis is defined as, to automatically allocate each spatial unit i.e. a pixel in a digital image into one of the several spectral classes (or, information classes) of interest present in the image on the basis of a multivariate vector of feature measurements $\mathbf{x} = (x_1, x_2, \ldots, x_p)'$ available for each such unit, where $x_1, x_2, \ldots, x_p$ are the brightness values of the pixel $\mathbf{x}$ in spectral bands $1, 2, \ldots, p$.

Numerous classification algorithms have been developed since 1936 when Fisher (1936) first employed the linear discriminant analysis (LDA) techniques to differentiate between the three species of Iris flower. These algorithms range from visual interpretation of printed images to advanced machine learning algorithms that imitate human learning behavior. Unlike visual interpretation or photointerpretation as discussed above in which the knowledge about the whole image is needed, automated classification methods only require information about a subsample of the image and hence are time and cost efficient. Based on the type of data, model of the data and the expected outcomes of the analysis, the automated classifiers can be broadly categorized into various types as depicted in Figure 1.2.

The two umbrella categories of classifiers in the image analysis literature are *supervised* and *unsupervised* classifiers which further can be categorized as parametric or non-parametric based on the distributional assumptions, and as hard or crisp classifiers based on the number of outputs for each spatial unit. The subsequent sections of this chapter discuss various unsupervised, supervised, parametric and non-parametric classifiers available in the literature. We exclude the whole body of literature on soft or fuzzy classifiers as these are beyond the scope of this thesis.

1.2.1 Unsupervised classification techniques

When the class labels of an image are not known apriori, unsupervised classification methods are employed for quanititative analysis of the images. By applying the unsupervised (clustering) algorithms, researchers hope to discover unknown, but useful, classes of items. Based on the fact that pixels within a

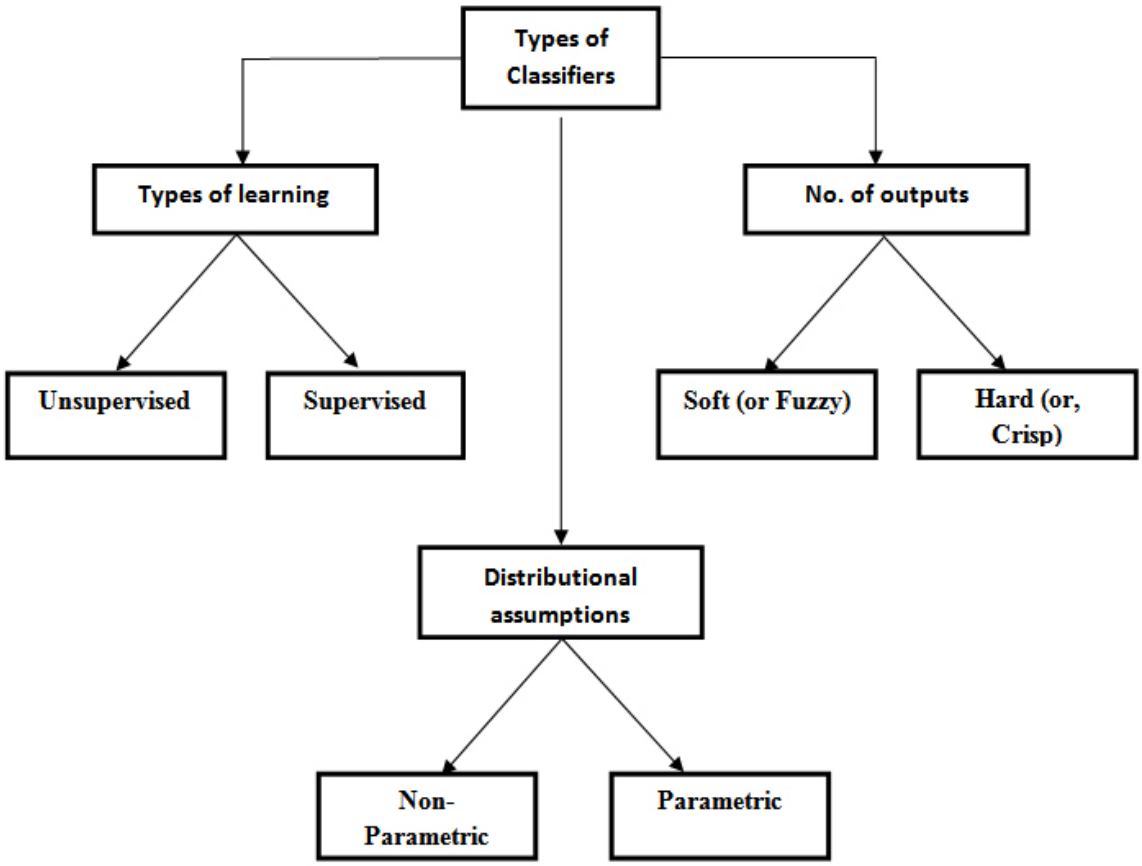
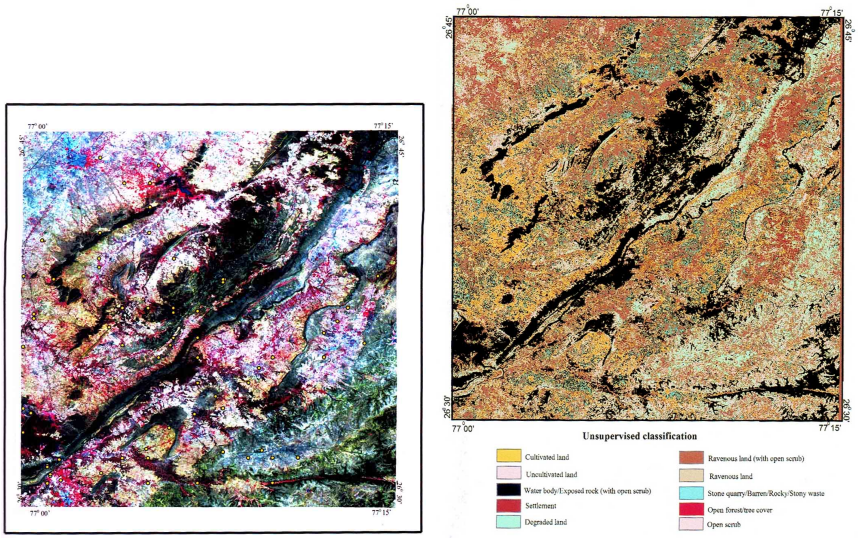


Figure 1.2: Types of classifiers.

group have distinct spectral signatures (Jain et al., 2000), unsupervised classification algorithms identify natural groups or clusters in the image using statistical clustering techniques and establish the relationship between the feature vectors and these statistical clusters. The process of determination of the number of clusters and identification of spectral clusters in the image in unsupervised classification approach operates almost independently and requires least human intervention. Figure 1.3 depicts the *false colour composite* (FCC) of a landcover satellite image and the classified version of it obtained through unsupervised classification of the image (Rais, 2015). Here, different colours indicate the 10 distinct clusters obtained through unsupervised classification of the FCC image. The most significant statistical unsupervised classification algorithms used in digital image anlaysis are ISODATA k-means algorithm (Ball and Hall, 1967; Mather, 2004).



(a) False colour composite image (b) thematic map of classified image

Figure 1.3: Unsupervised classification of a satellite image (Adapted from Rais (2015)).

1.2.1.1 k-means clustering

The k-means algorithm is an iterative optimization clustering algorithm, also referred to as migrating means algorithm based on the isodata algorithm given by Ball and Hall (1965). It works by iteratively migrating a set of cluster means using a *closest distance to mean* approach. Assuming the initial number of clusters in the image to be known, the k-means algorithm determines the location of, say k cluster means within the feature space by using a pre-defined set of feature vectors. Each spatial unit/pixel of the image is then assigned to the closest cluster mean as shown in Figure 1.4. The distance between the spatial units and the cluster centres are generally calculated using the Euclidean distance measure (d_E) or the Mahalanobis distance measure defined as

$$d_E = \sqrt{(\mathbf{x}_i - \boldsymbol{\mu}_j)^2} \qquad i = 1, 2, \ldots N; j = 1, 2, \ldots k \tag{1.3}$$

$$d_M = (\mathbf{x}_i - \boldsymbol{\mu}_j)' \Sigma^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_j) \qquad i = 1, 2, \ldots N; j = 1, 2, \ldots k \tag{1.4}$$

where, $\mathbf{x}_i$ is the observed feature vector of the ith spatial unit, μ_j and Σ are respectively the center (mean) and the variance-covariance matrix of the jth cluster. At the next iteration, the algorithms recalculates the center of each cluster based upon the the pixels allocated to it at the previous iteration and repeats the allocation procedure by finding the nearest cluster mean to the individual pixels or spatial units. This process of calculating the cluster means

and allocating the pixels to nearest clusters based on the Euclidean distances continues until the location of cluster means are unchanged or some predefined threshold by the user is reached.

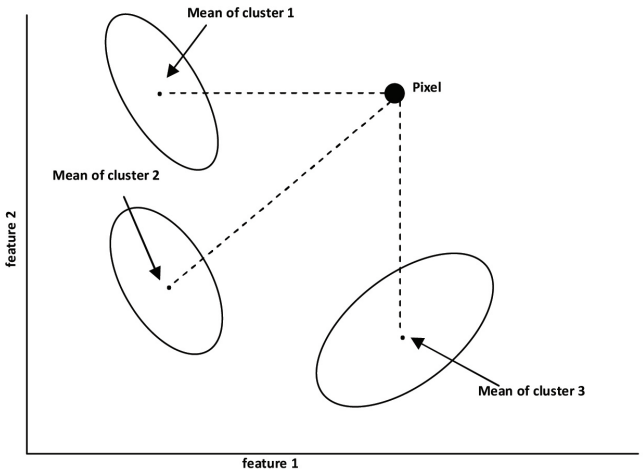


Figure 1.4: k-means clustering with Mahalanobis distance measures.

As noted in Tso and Mather (2009), use of Mahalanobis distance measure for calculating the distances between cluster centers and the pixels results in ellipsoidal clusters as d_M takes into account the shape of the frequency distribution for a given cluster, whereas d_E assumes perfectly correlated equivariant features for a given cluster and hence, results in circular clusters. It is because of this homogeneity assumption that Euclidean distances take ambiguous allocation decisions when d_E of a pixel is same from multiple clusters.

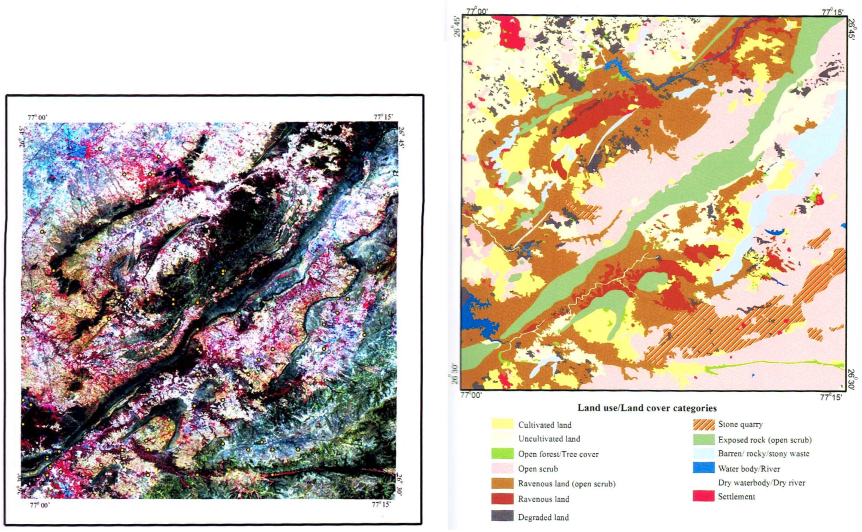
Although simpler in nature, the performance of unsupervised classification algorithms has been found to be inferior to the supervised ones due to the complexity of data in real world problems which might not always be easily separable in terms of the spectral signatures. Also, the computational load of calculating distance measures for each pixel increases as the dimensionality and the size of the datasets increase. Hence, it might not be a suitable algorithm for high-dimensional and large datasets.

1.2.2 Supervised classification techniques

Supervised classification approach to classification is used when the labels of the information classes present in an image are known apriori. This approach is most often used with almost all the statistical and non-statistical classification algorithms. Irresepective of the particular decision rule used for defining boundaries between the classes, the supervised classification approach involves five essential steps as explained in Richards and Richards (2008) which are, identification of class labels in the digital image, selection of training data pixels from the image which are samples of known identity from the image, estimation of the parameters for each information class using the training datasets and using them for training the selected classification algorithm to identify class labels of unknown pixels in the image, using the trained classifier to label each pixel in the digital image and finally to assess the accuracy of the classification algorithm. Thus, the major difference between the unsupervised and the supervised classification approach is the use of training datasets for learning the parameters of classification algorithms in supervised approach. The last two decades have seen a growing number of automated supervised classification algorithms being applied to a large number of image classification problems. Thenkabail (2015) gives a fair account of various supervised and unsupervised classifiers applied to land-cover mapping across various spatial and temporal scales.

Figure 1.5 shows the FCC image and the thematic map obtained after supervised classification of a landcover satellite image (Rais, 2015). Here, each of the 13 classes as depicted by different clolours in the thematic map are initially known to the analyst. The yellow dots on the FCC image depict the training data points obtained from each of the pre-determined classes through field survey of the landcover. It can be noted here that the supervised classification of the FCC image of a landcover satellite image depicted in the Figures 1.3 as well as in 1.5 resulted in more accurate classification of the image with 13 land use/land cover classes as compared to the 10 clusters that were observed through unsupervised classification.

Selection of training samples which are proper representatives of the corresponding spectral classes is the crucial factor that determines the efficiency of supervised classifiers. Although, this procedure of selecting training samples may be tedious but still supervised approach is often preferred over the un-



(a) FCC image (b) thematic map of classified image

Figure 1.5: Supervised classification of a satellite image (Adapted from Rais (2015)).

supervised approach as it generally gives more accurate class definitions and hence, improved classification accuracies. To this end, the classification approaches are further divided into the parametric and the non-parametric ones.

1.2.3 Parametric classification techniques

The supervised parametric classification techniques assume that the observed feature matrices for each spectral class in the digital image come from a known probability distribution and make inferences about the parameters of the classes under this assumption. The parametric classifiers are conceptually simpler and statistically more powerful than their non-parametric counterparts and happen to be the best performing classifiers as long as the underlying class distributions satisfy the requirements of the assumed probability models (Richards and Richards, 2008; Tso and Mather, 2009; Fukunaga, 2013). Maximum likelihood classifier (MLC) is the most popular supervised parametric classifiers in the image analysis literature.

1.2.3.1 Maximum likelihood classifier

Maximum likelihood classifier (MLC) which is explained thoroughly in Chapter 3 is perhaps the most common classifer used across all fields for executing classification tasks. MLC is derived from Bayes decision rule where likelihood of each pixel belonging to one of the pre-defined set of classes is calculated un-

der the assumption of known probability distributions of the underlying data classes. And, the pixel is allocated to the class for which it has the maximum likelihood. Under the most general setting of MLC, the underlying information classes are assumed to be of the form of multivariate normal model. The most commonly used parametric distribution models in designing a MLC are Gaussian distributions. Further, depending upon the associated assumptions of equal and unequal covariance matrices for component classes, MLC is formulated as a Bayes-normal-linear rule providing linear decision boundaries and Bayes-normal-quadratic rule resulting in quadratic decision boundaries respectively. In addition to the most commonly used maximum likelihood estimators of the covariance matrices, several regularization techniques are also available to obtain robust estimates in case of small samples (Friedman, 1989)
The MLC assumes the uniform prior probabilities for each information class. However, Davis et al. (1978); Strahler (1980) and Tso and Mather (2009) suggest the idea of modelling the prior probabilities and conclude that suitable modelling of the prior probabilities based on collateral information, such as associating priors proportional to the prevalence to each class, can improve the classification accuracies of a MLC.

The applicability of MLC as a robust method for practical classification problem requires a thoughtful pre-processing assessment of several required data charcteristics. The multivarite testing of the data is essentially required for assessing the assumption of the specified model fit of the information classes, application of data based transformations in order to make data confirm to the required model, detection and treatment of outliers if found any, assessment of the homogeneity of the classes in order to choose between the linear and quadratic forms of the MLC are some of the items that need beforehand attention for utilizing the full potentials of this most popular classifier. Moreover, the effectiveness of MLC depends on the efficient estimation of the mean vector and the variance-covariance matrices of the information classes which in turn depends on the selection of appropriate and adequate training samples for each of the spectral classes in the image as discussed in Section 1.4. A thorough discussion on all these considerations can be found in Fukunaga (2013).

The performance of MLC is in general affected when the underlying spectral classes do not conform to any multivariate probabiltiy distribution model and in such situations, non-parametric approach to the classification problems draws the attention of the analysts.

1.2.4 Non-parametric classification techniques

Non-parametric classification methods, also referred to as the distribution-free methods, do not impose any distributional assumptions on the observed feature matrices and hence, are considered robust for a wide variety of class distributions as long as the spectral signatures of the information classes are distinct. A variety of non-parametric classifiers are available in the image analysis litertaure. Among the statistical non-parametric supervised classification algorithms, parallelepiped and minimum distance classifiers are the most frequently used ones.

1.2.4.1 Parallelepiped classifier

Parallelpiped classifier is perhaps the simplest supervised classification technique requiring minimum amount of information from the user for training in the form of histograms of individual features in the training data for each of the information class (Richards and Richards, 2008). The range of each of the features is specified by examining these histograms and these multidimensional feature ranges are used to create boundaries of parallelepiped-like sub-spaces for each class. The decision rule simply assigns a pixel to the parallelepiped in which its spectral value is found to be lying. Figure 1.6 displays the segmentation of the feature space into the parallelpipeds for a two-dimensional classification problem.

Although, simplest in nature, parallelepiped classifiers are the least preferred choices in image analysis literature due to their inefficient performance to the likes of complex data found in the field of image analysis. The parallelepipeds based on the range of individual feature vectors which segment the feature space, do not cover the whole of feature space as shown in Figure 1.6 and often have considerable gaps between them, thereby avoiding any allocations of the pixels lying in these gaps. Moreover, these classifiers do not take into account any prior information, if available, about the class memberships of the pixels into account, and also in situations of correlated feature may result in overlapping parallelepipeds thereby resulting in confusion in the decsion making process and hence, produce ambiguous or random allocations of the pixels.

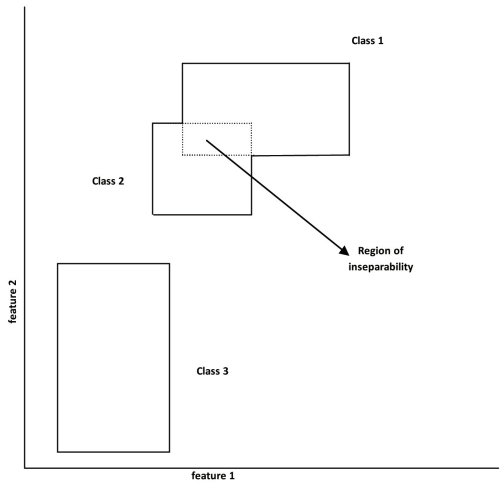


Figure 1.6: Parallelepipeds formation with the issue of overlapping parallelepipeds.

1.2.4.2 Minimum distance classifier

Minimum distance classifier (MDC) is another distribution-free simple statistical classifier which somehow resembles the k-means clustering algorithm in that it also calculates distance based dissimilarity measures for decision making. As the name suggests, this classifier calculates the distance between a pixel and the centroids of the training data classes and accordingly decides to assign the pixel to the nearest class. Although, other distance measures can also be used for calculating the distances between a pixel and the class centers, Mahalanobis distance and the Euclidean distance measures are most generally used in practice. The MDC is also commonly referred to as the Mahalanobis classifier when Mahalanobis distance measures are used for formulating the decision rule. Figure 1.7 shows the working of an MDC, using Euclidean distances for calculating the distances.This classifier is computationally efficient but is not theoretically as robust as the MLC and hence is often overlooked (Benediktsson et al., 1990).

Other non-parametric methods such as table-look-up classification, non-parametric density estimation, discussed in Richards and Richards (2008) and Fukunaga (2013) respectively, have also been proposed in past. But these are rarely used or even discussed as feasible alternatives in image classification literature. The reasons are, in the table look up approach to classification, a perfectly representative training data is required for constructing the look up table which might not be possible in practical applications. On the other hand,

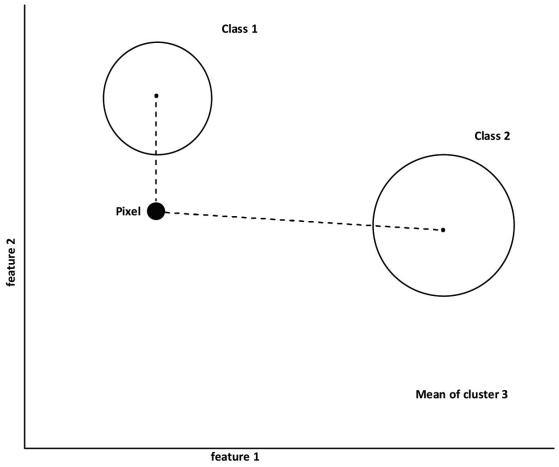


Figure 1.7: Working of minimum distance classifier with Euclidean distances. As shown, the pixel is at minimum distance from class 1 and hence is labelled as belonging to it by the minimum distance classification algorithm.

the non-parametric density approach often results in biased density and subsequently in biased Bayes errors estimates especially in a high-dimensional set up (Fukunaga, 2013) and hence, are often overlooked in comparison to other robust non-parametric alternatives discussed in Section 1.2.7.

1.2.5 Hybrid unsupervised/supervised classification approach

MLC is considered as the statistically most robust statistical classifier in the literature. However, it can only be used when the number of classes and their labels are already known to the analyst which is acheived through unsupervised clustering algorithms. A hybrid approach suggested in Fleming et al. (1975), brings together the strengths of both these approaches in order to overcome their respective disadvantages. This is termed as a *hybrid unsupervised/supervised approach* as reported in Richards and Richards (2008) and is often used in practice by the image analysts when the class labels are not known apriori. In this approach, first the image of interest is divided into information classes by using a clustering algorithm, say k -means clustering.

Training datasets are then obtained from these information classes and are supplied to the MLC for learning.

1.2.6 Advanced supervised classification methods

Although statistical classification approaches discussed in this chapter are the most commonly encountered classification methodologies in the image analysis literature but all of them have certain limitations. A thoughtful look on the literature of image anlaysis methods of last two decades shows an increasing fondness of the image analysts for advanced machine learning algorithms based on artificial intelligence and logic. A wide variety of studies justifying their superioirty over the fundamental statistical classifers have been published in the recent years (DeFries and Chan, 2000; Sebastiani, 2002; Pereira et al., 2009; Friedl et al., 2010). Artificial neural networks (ANNs), support vector machines (SVM) and classification trees (CTs) are the three most preferred machine learning methodologies in the field of classification. The main advantage of these classifers over the most popular statistical MLC is that they are distribution-free.

1.2.6.1 Artificial neural networks

Artifical neural network classifers based on artificial intelligence are one of the earliest machine learning techniques used in image analysis. The supervised ANN technique was first used for image classification by Benediktsson et al. (1990). And since then, studies from varied image analysis application fields, as discussed in Chapter 2 have demonstrated their effectiveness in image classification. ANN is a form of artificial intelligence that imitates some functions of human brain for complex non-linear problem solving. These networks construct decision boundaries by optimizing certain error criteria. These networks comprise of sequences of layers, each consisiting of simple processing elements called *neurones* fully interconnected to each other between these layers in a specified architecture. The potential discriminating power of ANNs has attracted a great deal of research for the development of various types of neural network architectures over the years (Lippmann, 1987). An increased number of hidden layers and the neurones per layer can increase the number of network parameters, time consumed and the chances of overtraining of networks. Hence, some regularization techniques like early stopping, addition of noise and early decay have been proposed in the literature. Apart from being distri-

bution free, ANNs boast of other elegant advantage over traditional statistical classifiers such as ability to process multi source data. Although, ANN classifiers have been shown to be significantly improving the classification accuracies over the traditional statistical classifiers even with smaller training datasets, the process of implementing these techniques is complex as compared to the statistical counterparts and hence, can be time consuming. Implementation, architectures and other considerations for the implementation of ANN have been discussed in detail in the next chapter.

1.2.6.2 Support vector machines

Support vector machines derived from statistical learning theory (Vladimir and Vapnik, 1995; Burges, 1998; Schölkopf and Burges, 1999) are one of the newest additions to the classification and regression methods. These methods work by fitting a separting hyperplane between the classes in the multidimensional feature space and try on maximizing the margins between the training patterns from each class and these hyperplanes, these training patterns being called as *support vectors*. The pioneering work of Gualtieri and Chettri (2000) for the classification of hyperspectral images has been followed by many researchers to analyze the theoretical properties and empirical performances of SVM applied to different kinds of classification problem. As compared to the statistical classifiers, SVMs are distribution-free methods which are more robust to noise, outliers and have greater generalization capabilities with smaller training datasets. On the other hand, user-defined parameters needed for learning SVMs may seriously restrict the performances of SVMs for complex solving classification tasks. Further details and a critical assessment of SVMs has been discussed in Chapter 2.

1.2.6.3 Decision trees

Decision trees refer to another class of some elegant distribution-free classifcation algorithms that have been widely used in the field of classification for solving pattern recognition problems and related tasks (Quinlan, 1987; Swain and Hauska, 1977). Decision trees predict class membership by recursively partitioning the p-dimensional feature space into more homogeneous subsets by a sequential method. Each node in a decision tree represents a feature of the pixel or the unit to be classified, and each branch represents a value that the node can assume. Iterative selection of the individual features that

are most salient is made at each node. Pixels are classified starting at the root node and are sorted based on their feature values. Since, at each node only those features that are needed for recognizing a test pattern are used, so feature selection is implicitly built in here (Jain et al., 2000).

Depending on the number of variables used at each step, there are univariate and multivariate decision tree (Friedl and Brodley, 1997). Multivariate decision trees are often more compact and more accurate than the univariate ones, but they involve more complex algorithms and hence can be affected by a suite of algorithm related factors (Friedl and Brodley, 1997). Moreover, decision trees can be designed manually based on user's expertize and knowledge about the data or through computer based automatic algorithms. Swain and Hauska (1977) proposed a heuristic search technique for solving more complex problems. Further contributions for designing computationally more efficient heirachical decision trees were made by Kulkarni and Kanal (1976); Kurzyński (1983); Lee and Richards (1985) and Kim and Landgrebe (1991).

The efficiency of the performance of a decision tree classifier heavily depends upon the nature of decisions being set for growing different branches of the tree and the sequence of attributes occuring within a tree. A thorough review of the various methods used to develop different types of decision tree can be found in Safavian and Landgrebe (1991). In the recent years, interest of the researchers has fast grown in the use of automatic methods of designing decision trees, such as ID3 (Quinlan et al., 1979), C4.5 (Quinlan, 1986) and classification and regression trees (CART) (Breiman et al., 1984) for the classification of complex and large size image datasets.

Their advantages over other approaches are that these are non-parametric methods which can effectively process both categorical and continuous predictor data, guarantees convergence of the algorithm even if the classes are not linearly separable, provides a more comprehensive understanding of relationships between the objects or pixels in the image datset and the output labels. One major limitation of these classifiers is the requirement of generation of rules for growing a tree which in turn requires expert knowledge of the area on the user's part. Chapter 2 of this thesis discusses some other aspects and variations of decision tree classifiers and conducts an empirical assessment of these classifiers.

1.2.7 Ensemble methods for classification

All the classifiers discussed till now in this chapter are the most popular classifiers in the field of image anlaysis and each one of them have their own *pros* and *cons*. Some are better in resolving one aspect of the classification problem while others may be better in other aspects and hence, researchers have shown interests in combining the potential of different classifiers in a single system for overall improvement in the classification outcomes. As noted in Jain et al. (2000), there are three main types of architectures for combining multiple classifiers, namely parallel, cascading and heirarchical (tree-like) with most of the combination schemes in the literature belonging to the parallel architecture where the results of independently invoked classifiers are combined by a combiner. In cascading, individual classifiers are invoked in a linear sequence whereas in hierarchical architecture, classifiers are arranged into a decision tree like structure.

Some recent studies on complex classification problem solving techniques using hyperspectral image data have led to two types of advanced combinational classification techniques (Crawford et al., 2003; Ham et al., 2005). First one is that of composite (or, hybrid) classifiers and the other one is that of ensemble classifiers. Composite classifiers are based on combining multiple individual methods usually in a stacked topology (Wolpert, 1992) and make use of combined expertise of the individually trained models to obtain an optimal classifier. Whereas, ensemble classifiers use a different approach where hundreds of classifiers are built and their decisions are combined usually by a weighted or unweighted voting method or more sophisticated method like consensus theory. Such a use of multiple classifiers may present a way out of the spiral of increasing complexities of the data. In general, though these classifiers present viable alternatives, one drawback is that they require to handle multiple learning algorithms at the same time resulting in an increase of processing complexity. The ensemble classifiers may avoid overfitting with noiseless data and reduce the variance and bias of the classification. Additionally, they may prove to be sensitive to noisy data and may involve larger computation times.

The effectiveness of these advanced classification techniques rely very much on the combining techniques. Several *combination methods* for amalgamating the outcomes of different decision rules have been proposed in the literature. Tso and Mather (2009) have discussed some of the significant combination methods such as, voting rules, Bayesian formulation, evidential reasoning and

multiple artificial neural network in detail with an application to the remotely sensed image data. A more detailed survey of such methodologies can be found in Tumer and Ghosh (1995); Kanellopoulos et al. (2012); Smits (2002); Briem et al. (2002) and Bruzzone et al. (2004). An essential requirement for combining the outputs from different classifiers for a classification task is that the individual component classifiers should be independent. This independence is acheived in practice either by using independent training feature sets for each classifier, using resampling techniques like stacking (Wolpert, 1992), boosting (Schapire, 1990) or bagging (Breiman, 1996a) or by using cross-validation methodology for estimating the errors (Breiman, 1996a).

1.2.8 Classification of hyper-dimensional imagery

Since, the emergence of hyperspectral sensor technology, issues related to the theoretical and experimental analysis of hyperspectral images have been explored (Lee and Landgrebe, 1993; Jimenez and Landgrebe, 1998). In hyperspectral imaging, the image is sensed with hundreds of spectral bands. This means that each individual pixel in the image is explained by a feature space of hundreds of dimensions. With such a large increase in the dimensionality of the feature space, many problems such as *Hughes phenomenon* (Hughes, 1968) arise in the path of the efficient classification of the data, particularly with the statistical classifiers. Most of the statistical classifiers require the estimation of the parameters of spectral/information classes and the number of training samples required for achieving robust estimates of such parameters is linearly proportional to the dimensionality of the feature space as noted in Fukunaga (2013). In turn, obtaining such large training datasets is not feasible in terms of cost, accessibility and time constraints in practical situations. Hence, with increased dimensionality, the regular classification methods may not be applicable to the raw data and the data is often treated first with dimensionality reduction techniques for improving the performance of statistical classification afterwards.

1.2.9 Feature reduction

Feature reduction, alternatively referred to as dimensionality reduction is a quintessential pre-classification step in image anlaysis of very high dimensional datasets. It can be achieved either by *feature selection* or *feature extraction* techniques. Feature selection is the method of employing techniques for se-

lecting a relevant feature subset of size say m from the complete feature set of size say p describing a pixel for reducing the dimensionality of the image data without discarding any meaningful information. Feature extraction on the other hand achieves dimensionality reduction by transforming the higher dimensional feature space to a lower dimensional one using linear or non-linear transformations.

Feature selection aims at assessing the discrimination capabilities of the reduced feature space using statistical distance measures. These feature reduction goals can be achieved using supervised as well as unsupervised approach. For feature extraction, linear transforms such as principal component analysis (PCA) (Landgrebe, 1980; Lennon et al., 2001), factor analysis, pursuit projection, decision boundary feature extraction (DBFE) (Lee and Landgrebe, 1993), discriminant analysis feature extraction (DAFE) (Jimenez and Landgrebe, 1998) are some of the most effective dimensionality reduction methods when the training set is sufficiently large in order to provide minimum number of transformed features. The literature also discusses some of the effective non-linear feature transformation techniques. The most widely used ones are Kernel PCA (Haykin, 1999; Schölkopf et al., 1998), multidimensional scaling (Sammon, 1969; Niemann, 1980; Borg and Groenen, 2005), neural networks (Lerner et al., 1999; Karhunen et al., 1997; Hyvärinen and Oja, 2000), self organizing maps (SOM) (Kohonen, 1995). But a disadvantage of these transform based dimensionality reduction methods is that they alter the original interpretation of feature space as discussed in (Kaewpijit et al., 2003).

The notables one among the strategies used in literature for searching the best subset of features are sequential forward search (Kittler and Kml, 1978; Pudil et al., 1994) and the method of steepest ascent (Serpico et al., 2007). The distance metric based feature selection methods used in literature are regression methods discussed in Gill et al. (1991); Han et al. (2001), instance based methods using inverse Euclidean distance weighting of k-nearest neighbours as discussed in Witten and Frank (2005). An other class of feature reduction methods is based on spectral similarity measures. The bands/features which are deemed as similar by some distance metric are replaced by the best representative sample. Spectral angle mapper (Kruse et al., 1993), Jensen Shannon divergence (Rao, 1982), Kullback-Liebler divergence measure (Kullback, 1959), information theory based measures (Maes et al., 1997; Guo et al., 2008) are some of the most popular measures used for detecting the spectral similarity between two spectral bands.

Several other advanced feature selection methods have been discussed in Serpico et al. (2007); Huang and He (2005); Huang and Wang (2006) and Su et al. (2011). Although, a number of advanced feature reduction methods have been proposed in literature, PCA continues to be the most readily used among the analysts due to its easy interpretation and availability in almost all off-the-shelf image analysis softwares.

1.3 Assessment of Classification Accuracy

The assessment of classification accuracy at the completion of a classification task allows a degree of confidence to be attached to the results. Also, with the advent of more sophisticated and advanced image classification methodologies alongside the previously available traditional but theoretically robust statistical methods, the necessity of performing an accuracy assessment for choosing the best suited method for a particular classification task has received renewed interest. Thus, we can say that there are two objectives for assessing a classifier's accuracy as noted in Hand (1997), to determine an absolute measure of quality of performance of a classifier or to compare the performance of multiple classifiers, in order to choose the best suited one for the problem at hand. Hence, it is necessary that an image analyst or a researcher should have a well equipped knowledge of both the factors needed to be considered while performing accuracy checks as well as the suitable techniques used for assessing the performance of a classifier (Congalton, 1991). A number of analysis techniques have been suggested in the literature for the purpose of assessment. A careful analysis of the problem at hand and its objectives is required before selecting one of these methods for evaluating a classifier. Hand (1997) provides a detailed thoughtful discussion on some classical evaluation techniques and the various aspects of assessing a classifier's performance. Here, we discuss some of the most widely promoted and used in image analysis literature.

1.3.1 Error matrix

An error matrix which is more popularly referred to as a *confusion* matrix in the image anlaysis field is the most elementary tool for reporting the classification accuracies and acts as an appropriate begining for many other analytical statistical assessment tools described in the subsequent sections. Table 1.1 reports some of the most general statistics that can be calculated from a (2×2)

confusion matrix. A confusion matrix is an $(m \times m)$ square array square array of numbers set out in rows and columns which express the cross classification of the predicted class labels by the true class labels of the sample units (i.e pixels). The columns in a confusion matrix correspond to the reference data points and the rows correspond to the labels assigned to these data points by a classifier. As shown in Figure 1.8, the confusion matrix depicts m data classes, $C_1, C_2, \ldots, C_m$ with n_1, n_2 and n_m number of validation data points respectively in them. The values on the main diagonal of the confusion matrix, $a_{ii}, i = 1, 2, \ldots m$ correspond to the correctly classified observations by the classifier and the off diagonal quantities, $\{a_{ij}, i = 1, 2, \ldots, m; j = 1, 2, \ldots m; i \neq j\}$ represent the number of observation in class j misclassified to class i by the classifier. Thus, in a confusion matrix the row sum, here $r_i, i = 1, 2, \ldots, m$ represents the total number of observations classified to the ith class by the classifier and the column sums, $c_i, i = 1, 2, \ldots, m$ represents the number of observations actually belonging to the ith class. A confusion matrix gives a summary of several indices of a classifier's performance. The overall classification accuracies of a classifier are obtained from the confusion matrices by dividing the sum of the main diagonal quantities by the total number of classified data points.

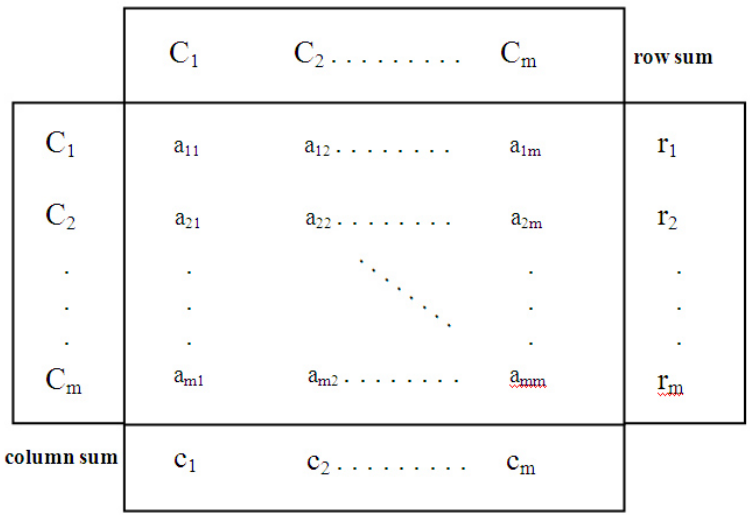


Figure 1.8: An $(m \times m)$ confusion matrix.

1.3.2 Misclassification error rates

By far the most popular measure for evaluating a classifier's performance is the *error rate* or *misclassification rate*. This is calculated by simply measuring the overall proportion of objects misclassified by a classifier which can be easily obtained from a confusion matrix as explained above. However, superficially estimating these error rates using the training datasets or the datasets which are used for learning the classification rule typically lead to an underestimate of the future error rates, which are obtained by calculating misclassification proportions of unknown samples. This is beacause the classification rule will be optimized in some sense for the training datasets while learning process. However, this problem can be minimized by using larger training datasets. But, this might be a practical limitation for many real life image datasets as found in in the field of medical diagnosis where one might not have the availability of a large number of known training datasets. Error rates obtained using training data are called as *substituion* or *apparaent error rate* (APER).

This problem of overfitting of the training datasets and overestimation of error rates can be solved by dividing the total available known samples into separate training datasets which are used for learning the classification rule and the *test dataset* which is used for evaluating the rule. Hence, obtaining the error rates from the test dataset provides an unbiased estimate of the actual performance of the classifier on future samples. The error rates estimated using separate test datasets are called as *actual error rates* (AER).

It may not be feasible to use the independent test set approach for estimating actual error rates when the complete available dataset of known pixel values is not sufficiently large. In such situations of limited data availability, a compromised approach called as *cross-validation* can be used for estimating the apparent and actual error rates. This approach involves extracting the subsets of the whole known sample dataset to test the performance of classifiers trained using the remaining sample data points, then repeating this for different subsets and averaging the results. This methodology can be applied with three variants (Hand, 1997):

- *k-fold*: In this type of cross-validation, k-mutually exclusive subsets of the known sample dataset are defined, each one being used in turn as a test set for the classifier built on the remaining $(k-1)$ subsets.

- *leave-one-out*: This variant being the most popular was introduced by

Lachenbruch and Mickey (1968). Out of the total, say n, known sample data points, a single point is used a test set for the classifier trained using the remaining $(n-1)$ datapoints as the training set. And, hence the whole process is repeated n times for obtaining averaged unbiased estimates of error rates.

- *bootstrap*: In this method, the potential of bootstrap sampling is used for efficient and unbiased error estimation. Here, a random sample of size n, which is the size of the complete known dataset, is taken with replacement as the training dataset for building the classifer and finally the complete dataset is used for testing the classifier's performance in terms of the misclassification rates.

The error rates described above give an estimate of the overall performance of a classifier and do not provide specific information about the error rates of individual classes. These overall error rates may prove to be misleading in situations when the *prevalence* (unequal class size) of the various classes involved in the data are significantly different or when the *cost of misclassifications* from various classes need to be considered (Tso and Mather, 2009). For example, if an image is classified into two classes, and the two classes cover the image area in the ratio of $2:1$, then this is called as the problem of prevalence as the first class is more prevalent in the image data than the second one and in such situation the classifier gets biased resulting in lower misclassifications in the prevalent class. Secondly, in the field of medical imaging where the imagery is used for diagnosing a subject into patient or non-patient category, the cost of a misclassification may be very risky.

In the similar situations, it is advisable in the literature to assess the classifier's performance in terms of the error rates of individual classes called as the ommission and commission errors. *Errors of ommission* (e_O) correspond to those pixels belonging to the class of interest in the image which the classifier fails to recognize. These are calculated by dividing the total number of pixels in a class misclassified to other classes by the total number of reference pixels (or, the ground truth pixels) in that class and are also referred to as measures of *producer's accuracy*. From the confusion matrix shown in Figure 1.8,

$$e_{Ok} = \sum_{j} a_{kj}/c_k \qquad \forall j; j \neq k \tag{1.5}$$

is the ommission error for the kth class. Whereas, the *commission error* (e_C)

of a class correspond to those pixels from other classes that the classifier labels as belonging to the class of interest. These are calculated by dividing the total number of pixels incorrectly misclassified in a class divided by the total number of pixels that were classified in that class and is often referred to as the measure of *user's accuracy*. From the confusion matrix,

$$e_{Ck} = \sum_{i} a_{ik}/r_k \qquad \forall i; i \neq k \tag{1.6}$$

is the commission error for the kth class.

1.3.3 Agreement measures

In classification problems the risk associated with a classifier can be defined as the misclassification rate. Since the risk is seldom deterministic, we might also need to check the reliability of the risk estimation. This is achieved using the random agreement measures. This section refers to a class of discrete multivaraite techniques which take into account the chance allocations made by a classifer for assessing its performance. The most popular among them is the *kappa* measure which was suggested by Cohen (1960) and is therefore often referred to as Cohen's kappa. The kappa measure is a more powerful technique for evaluating a classifier as it uses all the information in an error matrix, unlike the previously discussed error rates which consider either the principal diagonal elements or, the off-diagonal elements only. It provides a better measure of the accuracy of a classifier than the overall accuracy as it consider inter-class agreement and is calculated as

$$K = \frac{p_0 - p(k)}{1 - p(k)} \tag{1.7}$$

where, K is the kappa coefficient, p_0 is the proportion of overall agreement and p(k) is the chance agreement probability defined by Cohen as

$$p_0 = \sum_{i=1}^{m} \frac{a_{ii}}{N} \tag{1.8}$$

$$p(k) = \sum_{i=1}^{m} \frac{(r_i \times c_i)}{N^2} \tag{1.9}$$

where, N is the total number of observations. The higher the value of Kappa coefficient K, the better the classification performance. In the ideal situa-

tion, when all the pixels are correctly classified, it takes value equal to 1. The kappa coefficient has many attractive features as an index of classification accuracy. In particular, it makes some compensation for chance agreement and a variance term may be calculated for it enabling the statistical testing of the significance of the difference between two coefficients (Rosenfield and Fitzpatrick-Lins, 1986). Frequently, there is a desire to compare different classifications and so matrices. To further aid this comparison, some have called for the normalization of the confusion matrix such that each row and column sums to unity (Congalton, 1991; Smits et al., 1999). Smits et al. (1999) argued that kappa coefficient should, in some circumstances, be adopted as a standard measure of classification accuracy.

Although, Cohen's kappa coefficient is often used for assessing the level of chance agreement in the ouput of a classifier, it is found to be sensitive to both prevalence as well as to bias (Byrt et al., 1993; Gwet, 2002). Gwet reported an alternative statistic in Gwet (2014), named as AC1 statistic to estimate the chance agreement of a classifier defined as

$$AC1 = \frac{p_0 - p(\gamma)}{1 - p(\gamma)} \tag{1.10}$$

where,

$$p(\gamma) = \frac{1}{(m-1)} \sum_{i=1}^{m} \pi_i (1 - \pi_i) \tag{1.11}$$

is the chance agreement probability with $\pi_i = (r_i + c_i)/2N$. Gwet's AC1 statistic is found to be more robust to the effect of prevalence of classes (Wongpakaran et al., 2013). Foody (1992) in his study noted that Cohen's kappa coefficient overestimates the chance agreements and underestimate the accuracy and hence, suggested an alternative formulation. It should be noted here, that the reliability of chance agreement measures depend upon the the test data. And hence, studies suggest that test data should be appropriately chosen with simple random sampling (Janssen and Vanderwel, 1994; Stehman and Czaplewski, 1998; Foody, 2002; Congalton and Green, 2008) in order to get maximum benefit from kappa measures.

1.3.4 Area under ROC curve (AUROC)

The Receiver Operating Characteristic (ROC) curve has long been used as a good way of visualising a classifier's performance in order to select a suitable

Measure	**Formula**
Overall classification accuracy	$(a+d)/N$
Overall misclassification rate	$(b+c)/N$
True positive rate	$a/(a+c)$
False positive rate	$b/(b+d)$
Prevalence	$(a+c)/N$
Sensitivity	$a/(a+c)$
Specificity	$d/(b+d)$

Table 1.1: Accuracy measures calculated from a (2×2) error matrix. The two classes being referred to as positive class and the negative class. Here a, b, c, d, N are the components $a_{11}, a_{12}, a_{21}, a_{22}, N$ respectively of the error matrix as described in Figure 1.8.

operating point, or decision threshold (Hand, 1997) in a two class problem. One of the earliest adopters of ROC graphs in machine learning was Spackman (1989), who demonstrated the value of ROC curves in evaluating and comparing algorithms. Recent years have seen an increase in the use of ROC graphs in the machine learning community due to the realization that simple classification accuracy may often be a poor metric for measuring performance (Provost and Fawcett, 1997; Provost et al., 1998). In addition to being a generally useful performance graphing method, they have properties that make them especially useful for domains with skewed class distribution and unequal classification error costs. These characteristics have become increasingly important as research continues into the areas of cost-sensitive learning and learning in the presence of unbalanced classes.

An ROC plot is a two-dimensional graph obtained by plotting all *true positives* on the y-axis against their equivalent *false positives* (refer to, Table1.1) for all available thresholds on the x-axis. Figure 1.9 shows ROC curves for four classification rules labelled A through D. The classifiers with an ROC curve which follows 45 is termed as useless as it classifies equal cases in both the classes and hence, would not separate the classes at all. Whereas, an ROC curve for a perfect classifier would follow both the axes classifying all the positives in the positive class. As such dominance relationships between classifiers are studied using ROC curves however, when comparing a number of different classification schemes it is often desirable to obtain a single figure as a measure of the classifier's performance. The area under the ROC curve (AUC) is usually taken to be an important index for this purpose because it provides a single measure of overall accuracy that is not dependent upon a

particular threshold (DeLeo, 1993; DeLeo and Rosenfeld, 2001). The curve which dominates the other sweeps larger such area under it and corresponds to the better classification rule. For example, in Figure 1.9 clearly the curve corresponding to the classifier C is better as it sweeps a larger AUC.
The AUROC is a single discriminability measure for two-class scenario. Generalization approaches for calculating AUROC in multi-class problems have been discussed in Hand and Till (2001) and Provost and Fawcett (2001).

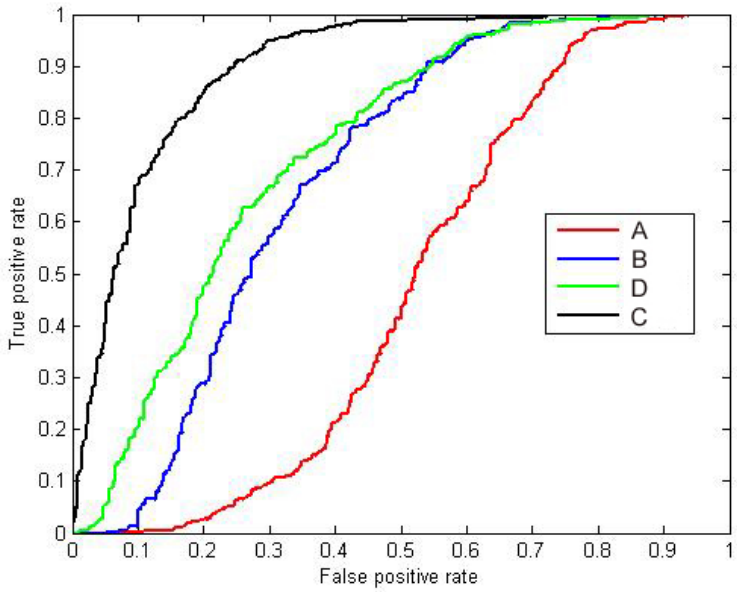


Figure 1.9: ROC plot for four classifiers labelled as A, B, C and D.

Apart from these measures, there may be other ways for calculating the predictive accuracy of a classifier such as Kolmogorov-Smirnov (KS) statistic, likelihood ratios, pairs of measures such as specificity and sensitivity or precision and recall, measures of accuracy of probability estimates such as Brier or log score, and many others. A detailed account of these methods can be seen in some significant works like Aronoff (1982, 1985); Kalkhan et al. (1995); Koukoulas and Blackburn (2001); Piper (1983); Rosenfield (1981); Rosenfield and Fitzpatrick-Lins (1986); Huberty (1994); Flach (2003); Hand (1997); Vieira and Mather (2001) and Pepe (2003). Thus to conclude, the choice of the most appropriate classification assessment rule must be made on the basis of some objective considerations as suggested in Fielding and Bell (1997).

1.4 Data Considerations for Supervised Statistical Classifiers

When using supervised learning approach specifically with a parametric formulation of problems in classification of digital images, there are a number of factors that need the anlayst's attention before the final version of the classifier is constructed and applied to the problem. Some of the most elementary ones are discussed in this section.

1.4.1 Sampling scheme

Both training and test samples characterizing the classes of interest in an image are required to be selected for performing supervised classification of the image. A sampling scheme describes the way in which these samples are drawn from the whole digital image dataset. There are certain restrictions on sampling, including cost, availabiltiy of source of information such as maps, photographs, and accessibility, size of the area of interest, temporally changing nature of the data. All such factors make it difficult to conduct a thorough and statistically valid sampling procedures for extracting known samples from the image. Hence, attention must be paid to all such factors while selecting an appropriate sampling scheme in classification tasks, particularly when statistical classifiers are employed as performance of these classifiers depend significantly on the training samples for the fitting of the assumed probability models (Tso and Mather, 2009).

Many sampling schemes have been suggested in the image analysis literature. Congalton (1988) suggested that both simple random sampling and stratified random sampling work satisfactorily in classification tasks. Whereas, Atkinson (1991, 1996) notifiy that these standard statistical sampling rules, do not work well with spatial data (as found in remote sensing) where locations are fixed resulting in autocorrelated features and hence he proposed geostatistical methods for such situations. Detailed theory on these methods can be found in Curran (1988); Woodcock et al. (1988b,a); Van der Meer and De Jong (2011). Wang et al. (2005) suggested a semivariogram technique for determining the optimal sampling space. Moreover, when it comes to assessing a classifier using Kappa measure, simple random sampling must essentially be used in order to satify the required assumptions (Stehman, 1992; Congalton and Green, 2008). Apart from these, cluster sampling can also be used which

allows the collection of a large number of samples. However, large cluster samples i.e. having more than 10 pixels are not recommended in the literature due to the autocorrelation effect (Congalton, 1988).

1.4.2 Sample size

Appropriate sampling scheme is not the only criterion that should be considered while designing a classifier. Sample size considerations are equally important for acheiving specified levels of classification accuracy and for obtaining statistically valid measures of these accuracy. The issue of appropriate training sample sizes has been widely discussed in the literature by several authors (Hord and Brooner, 1976; Van Genderen and Lock, 1977; Rosenfield et al., 1982; Mather and Koch, 2011). Congalton and Green (2008) suggest a method based on the multinomial distribution for estimating sample size per class. Based on the notion used in univariate statistics, Mather and Koch (2011) suggested that the training sample size per class should be 30 times the number of parameters to be estimated or alternatively 30 times the number of features used for classification.

This suggestion by Mather gives satisfactory results in most of the cases as far as the dimensionality of the feature space is not too high. But as the dimension of the data increases as in hyperspectral imagery, the precision of the estimates obtained from samples of fixed size suggested in Mather (2004) becomes substantially low, thereby decreasing the efficieny of parametric classifiers such as MLC which operate by defining the model of the data distribution. In such situations, unfeasibly large datasets are required for efficient estimation of parameters which may not be possible to obtain in practical situations and hence, feature extraction methods or dimensionality reduction methods will be required to analyze the data. Foody and Mathur (2006) suggested the adoption of four elementary principles, namely, selection of the most informative training samples, acceptance of imprecise descriptions for spectrally distinct classes, selective class, exclusion and adoption of a one-class classifier for considerably reducing the training size without significantly affecting the required accuracy.

Although, all of these above discussed sampling schemes and sample size considerations are directed towards the use of statistical classifiers, non-parametric supervised classifiers such as ANN, SVM and DTs are also found to be affected by the sampling scheme and sample size considerations (Evans, 1998).

Although, these classifiers are non-parametric in nature and do not require the estimation of class parameters form the training samples, large enough training datasets that can represent the characteristics of each class are still required for efficient learning of these distribution-free methods of classification. Out of ANN, SVM and RF classifiers, SVMs have been found to be robustly performing even with small training data size in Foody (1996), Foody and Mathur (2004) and Foody and Mathur (2006).

1.4.3 Adequacy of training data

Along with estimating appropriate sample sizes, extraction of adequately representative training data in the sense that no erroneous data points are included in the training dataset also needs thoughtful attention of the user. Such erroneous data points commonly referred to as outliers may significanlty alter the parameter estimates required for modelling of the class distribution in MLC. However, if even, after all precautionary measures, such sample points find their way into the training data, they should be removed or treated accordingly. Such outliers can be accomodated by the robust statistical estimation of the parameters using weighting methods (Mather, 2004). Their effect on classification accuracies can be further reduced by using cross-validation approach for accuracy assessment or by using ensemble classifiers (Brodley and Friedl, 1999).

2

Performance of Non-Parametric Classifiers on highly skewed data

2.1 Introduction

Pattern recognition or specifically classification is the problem of allocating an unknown object based on a set of features into one of the several possible classes (or populations). These features can be thought of as p- dimensional vectors of measurements describing the object. The object to be classified is a broad term which may specifically denote a pixel or a set of pixels for digital image classification or can be a patient in the field of diagnostics or a waveform in the field of speech and voice recognition or it can also be the measurement of a subject's response in physochological tests. Classifiers can be broadly classified into supervised and unsupervised classifiers depending upon the type of learning. Further, the classifiers can be termed as parametric and non-parametric classifiers depending upon whether any distributional assumptions are imposed on the underlying classes or not. The supervised classifiers are required to be trained by samples of known identity referred to as training datasets and hence need intervention of human expertise for obtaining these training samples whereas the unsupervised classifiers are completely dependent on the algorithm used for clustering observations of similar characteristics into an information class. Although, unsupervised classification algorithms are fast and easy to implement as they do not require to be trained but still the supervised classification techniques are more preferred by analysts and researchers

as they are able to produce more accurate results in terms of lesser misclassification errors and are also equally efficient in terms of fast computations (Tso and Mather, 2009). Hence, we deal with only supervised classifiers in this thesis. From now on in this thesis wherever we talk of a classifier we mean supervised classifiers only. With time and wide spread availability of the *state-of- the-art* computational techniques, non-parametric classifiers have emerged as strong competitors of the conventional parametric classifiers. In this chapter, we highlight the shortcomings of the parametric classifiers which ultimately create the scope for the non-parametric classification techniques. Additionally, within the group of non-parametric classification techniques we highlight, discuss and compare the performance of, the most recent machine learning classifiers in handling significantly positively skewed datasets. The chapter is divided as follows, remaining part of the Section 2.1 discusses the motivation behind the investigation carried out in this chapter and the objective of the investigation, Section 2.2 highlights other comparative works carried out in past with the machine learning algorithms taken up for study in the present chapter, Section 2.3 gives a brief discussion of the methods and the classifiers used for comparison in the present work. A detailed investigation on the comparative performances of the non-parametric classifiers for real and skewed simulated data is given in Section 2.4 and the conclusions have been discussed in Section 2.5.

2.1.1 Motivation

Discriminant analysis techniques based on parametric classifiers are the most readily and widely used techniques for classifying an object (or observation) into one of the several possible classes (or populations). These parametric classifiers which are based on the assumption that the populations in the feature space come from some theoretical statistical probability distribution are found to perform optimally when the required assumption of known probability distribution is fulfilled by the underlying data classes. Maximum Likelihood Classifier (MLC), minimum distance classifiers, K- means clustering (KMC) are some of the most popular parametric classifiers. Among them MLC or the Bayes classifier which assumes Gaussian distribution for the underlying population is the most frequently used supervised parametric classifier in the field of classification or discriminant analysis. The MLC classifies an observation into the population for which it has the maximum likelihood and hence requires the

estimation of the parameters of the assumed multivariate normal distribution for different classes which in turn will need a proper representative sample of the data classes. However, the normality assumption of the MLC is often found to be violated in real life situations and the real datasets are generally found to be skewed in nature. For example, in complex land cover classification problem, if area under crop is one of the several land use categories then the presence of trees along with crops as is the case in agro forestry or the presence of stressed crops will result in the skewed spectral feature distribution of the crop class as healthy crops, stressed crops and trees will have different spectral signatures. Similarly, the class representing water bodies in land cover satellite image will appear to be skewed in the presence of different types of water bodies in a single image, for example clean water, turbid water, water containing chlorophyll which represents the presence of algae colonies or industrial waste water. In face recognition problems also when the with-in class variability is larger than the between class variability introduced by changes in illumination, the data is found to be skewed in nature (Zhang and Jain, 2006). In all such situations the distributions of the underlying data classes will deviate from normality and will exhibit skewed nature. This limitation of distributional assumption poses a threat to the efficient performance of the MLC when the data is non-normal in nature.

In some benchmark studies like Lachenbruch et al. (1973); Clarke et al. (1979); Beauchamp et al. (1980); Baron (1991) and Khondoker et al. (2013), the authors tested the robustness of the MLC and studied the extent to which the performance of MLC can be affected by various types of non-normality of data using simulations, real datasets and graphical methods as well. All of these studies concluded that the performance of MLC was found to be sensitive to deviations of the data classes from the assumption of multivariate normality and consequently the conventional MLC is not expected to perform optimally when the data classes are significantly skewed in nature. Hence, in the presence of such non-optimal situations for the conventional MLC, the researchers and experts suggest to look out for the alternative non-parametric classifiers which are free of any distributional assumptions and hence are expected to perform well with a variety of distributions as long as the class signatures are reasonably distinct.

After a comprehensive review of the literature of classification techniques discussed in Section 2.2 we found that although a number of studies have been conducted for comparing the performances of the parametric MLC with its

non-parametric counterparts for particular case based studies, nothing much has been done about the performance of the non-parametric classifiers when the data is severely skewed. Hence, in the present study we attempt to fill this gap by particularly focussing on the performance of the non-parametric classifiers for classifying severley positively skewed data.

2.1.2 Objective of the study

The performance of a classifier depends on many factors and in the absence of any particular guidelines for selecting the best classifier for a specific study (Lu and Weng, 2007), the need is to look out for that one algorithm which can fairly work with a larger number of datasets without compromising, significantly, with the classification accuracy. This limitation is not restricted only when one has to choose between parametric and non-parametric classifiers, but within the class of non-parametric classifiers we need to look out for the classifier which can perform fairly on a larger number of varied datasets. Hence, leaving the parametric classifiers to be discussed in detail in the next chapter, we dedicate this entire chapter to the study of non-parametric classifiers with an aim to zero in the most efficient non-parametric classifier for classifying significantly skewed datasets. In this chapter we elaborately discuss three recent and most advanced non-parametric classification algorithm i.e. Artificial neural networks, Support vector Machines and Random Forests and explore their ability in efficiently classifying the skewed datasets using extensively simulated skewed datasets as well as some real datasets.

2.1.3 Non-parametric alternatives to parametric classifiers

With the advent of faster and more sophisticated computing options and the culture of interdisciplinary research gaining acceleration, a good number of non-parametric machine learning algorithms based on statistical, logical, fuzzy ensemble and kernel based methods have been developed in the last few decades (Sahoo et al., 2012) to overcome the restrictions of imposing normality assumption on the underlying data classes as required by the traditional MLC and hence, to obtain more accurate classification results.

Among the non-parametric classifiers available, parallelepiped and minimum distance classifiers fall under the statistical classifiers category. Paral-

lelepiped classifiers are the simplest ones of all the non-parametric classifiers and require minimal information in the form of minimum and maximum values of all the feature in each of the classes which define the boundaries of the parallelepipeds and each observation is then checked if it lies in any of the defined parallelepipeds. This classifier is highly affected by the presence of overlapping parallelepipeds and inability of locating a new observation in any of the defined parallelepipeds and hence is not considered a robust choice for most of the classification problems. The second one i.e. the Minimum distance classifier calculates the distance between an observation and the centroids of the different training classes using the Mahalanobis distance measure and accordingly decides to allocate the observation to the class which is nearer to the observation in terms of lower value of the distance measure. This classifier is also found to be mathematically fast and does not include any complex underlying mathematical concepts but its performance has always found to be inferior to the more robust MLC (Benediktsson et al., 1990). Moreover, the performance of both of the above discussed classifiers is expected to be affected a lot by the presence of heterogeneity and outliers in the data classes which are the common characteristics of skewed datasets.

Thus, keeping in mind the limitations of these statistical non-parametric classifiers we turn our attention to the more advanced machine learning algorithms for classifying skewed datasets. Among the class of non-parametric machine learning algorithms, artificial neural networks, support vector machines and random forests classifiers have gained considerable popularity among the researchers and the analysts in the field of remote sensing, voice recognition, text classification, medical diagnosis of terminal diseases etc. The major advantage of these classifiers over MLC is that they neither assume any statistical probability distribution for the data classes nor require any statistical parameter estimation to separate the classes and hence guarantee better classification outcomes (Paola and Schowengerdt (1995); Foody (2002)), when the underlying data is not normal or specifically skewed in the context of this thesis. Apart from this, these classifiers are able to incorporate class-relevant categorical and continuous observations into the feature space, can tease apart complex feature spaces and are capable of performing many-to-one classification where multiple manifestation of the same category are present in the obseravtion matrix. Many works have been published in the recent years, discussed in the next section, which compare the performance ANN, SVM and Random Forest classifiers with that of the MLC. The discussion concludes that these

non-parametric classifiers should be preferred over MLC. In the next section, we discuss some comparative works on ANN, SVM and RF.

2.2 Background

Classification procedures are widely used in a variety of fields due to which a large number of studies comparing the performance of disfferent types of classifiers have been produced. Hence, for the sake of comprehensiveness and better understanding we give an account of some of the recent comparative works with respect to the fields in which they were conducted.

- *In remote sensing*: Huang et al. (2002) compared the performance of SVM with MLC, ANN and decision tree classifiers for the classification of a six band Thematic Mapper (TM) image and found SVM to be competitive enough with the other two methods. Erbeka et al. (2004) compared the performance of MLC with multilayer perceptron (MLP) and Linear Vector Quantization (LVQ) ANN classifiers for classifying a Landsat TM data and suggested the better performance of the ANN classifiers because of their ability to process multisource data easily. Kavzoglu and Kolkesen (2009) assessed the effect of kernel choice on the SVM classifiers and concluded that SVM classifiers based on rbf kernels outperform the MLC for the classification of landcover images. Otukei and Blaschke (2010) found decision tree classifiers to be performing better in general in terms of the classification accuracies than the SVM and the MLC classifiers for classifying Landsat TM datasets. Apart from these, Zhuang et al. (1995); Atkinson and Tatnall (1997); Cortijo and Blanca (1997); Michelson et al. (2000); Keuchel et al. (2003); Pal and Mather (2003); Lu et al. (2004); Olthof et al. (2004); Pal and Mather (2004) and South et al. (2004) are some other comparative works conducted for specific case based classification problem.

- *In bioinformatics and diagnostics*: Diaz-Uriarte and de Andres (2006) investigated the performance of Random Forest, Diagonal linear discriminat analysis (DLDA) technique, KNN and SVM classifiers for classifying microarray datasets and found RF classifiers to be performing exceptionally well as compared to other classifiers for very high dimensional microarray datasets. Dudoit et al. (2002) employed three microarray datasets for the classification of tumours and found DLDA and ANN

classifiers to be performing remarkably well as compared to more sophisticated aggregated or bagged decision tree classifiers. Statnikov et al. (2008) in their study on the microarray based cancer classification using a large number of datasets found SVM classifiers to be performing better than the RF classifiers with and without adopting any feature selection procedures. Khondoker et al. (2013) conducted an extensive simulation study to compare the performance of various significant classifiers i.e. LDF, SVM, ANN and RF under various settings of number of features, training sample size, correlation between the feature, and variability within the data. They concluded that different classifiers performed optimally under different settings. For example they found that LDF is superior for smaller number of correlated features and SVM for data with high dimensional feature sets. RF was found to be performing better in case of more variable data classes. Apart from these works other significant comparative assessments of the classifiers for microarray data classification for cancer diagnosis can be found in Tan and Gilbert (2003); Man et al. (2004); Lee et al. (2005); Huang et al. (2005); Dossat et al. (2007); Cutler et al. (2007b); Pirooznia et al. (2008); Boulesteix et al. (2008); Rocke et al. (2009); Yousefi et al. (2011b); Hanezar and Dougherty (2010); Demšar (2006) etc.

- *Other fields*: Apart from these two fields a number of comparative studies in the field of text recognition, speech recognition, ecology and financial data prediction have been produced. Zhang et al. (1999) employed ANNs for financial data prediction and established their superiority over logistic regression techniques. Cutler et al. (2007b) compared RF classifier with LDA, logistic regression, SVM and ANN classifiers for classifying three groups of organisms and for predicting invasive species presence and observed RF to be the best performing classifier in terms of higher cross validation accuracies.(Zhang et al., 1999) compared SVM and ANN for credit rating analysis. Tsai and Wu (2008) compared the performace of ANN with multiple classifier techniques for financial data prediction.

Majority of these comparative works are case based studies and investigated the comparative performances of the classifiers on particular but vastly varied real datasets, but not much has been done for assessing the performance of these non-parametric classifiers on the simulated datasets and specifically the skewed ones with few exceptions, like Aeberhard et al. (1994); Man et al.

(2004); Huang et al. (2005); Diaz-Uriarte and de Andres (2006); Hanezar and Dougherty (2010); Khondoker et al. (2013). However, the better performance of any particular classifier on one or a few instances cannot guarantee the same for all the other datasets, hence simulation studies may be a better alternative for objectively and feasibly comparing the performance of various machine learning algorithms (Yousefi et al., 2011a). Apart from being illustrated on the real datsets the results obtained in this thesis are based on simulated datasets as well, and hence, they do not specifically cater to the classification issues of any particular discipline and can be referred to in general for any type of classification problem.

2.3 Non-Parametric Classifiers and Other Methods Used

2.3.1 Artificial neural networks (ANNs)

Fascination of the researchers with understanding and emulating the efficiency of the eye-brain combination in processing large amount of data from varied sources led to the discovery of first neural network model in 1943, (McCulloch and Pitts, 1943). Artificial Neural networks comprise of a set of machine learning algorithms which use artificial intelligence techniques for complex problem solving. ANNs have evolved over the years as a robust pattern recognition alternative to other methods with contribution from varied disciplines ranging from neuroengineering, financial data prediction, quality control, modeling and prediction to pattern recognition. Detailed conceptual explanation of ANNs can be found in Haykin (1999) and Garson (1998). ANN classifiers enjoy pretty attractive advantages over other classifiers of being data driven self adaptive or distribution free methods capable of estimating posterior probabilities and handling multi-source data efficiently (Benediktsson et al. (1990); Richard and Lippmann (1991)). Additionally, they are hailed as universal approximators (Hornik, 1991) and non linear models and hence have been effectively applied across a wide variety of application fields which include bankruptcy ((Leshno and Spector, 1996; Zhang et al., 1999; Atiya, 2001)), medical diagnosis (Komori and Eguchi, 2015), product inspection, fault detection in industrial applications (Fukuda and Shibata (1992); Meireles et al. (2003)), handwriting recognition (Kalaichelvi and Ali, 2012), speech recognition (Dahl et al., 2012)

and bond rating. In contrast to all these pros, ANNs have some serious limitations. In order to make them perform efficiently ANNs should be trained with proper choice of network architecture and optimal parameters in the form of number of nodes and the number of hidden layers used for training the network (Kotsiantis et al., 2007). Elaboration on the limitations of ANN and other non-parametric machine learning algorithms will be discussed in the next chapter.

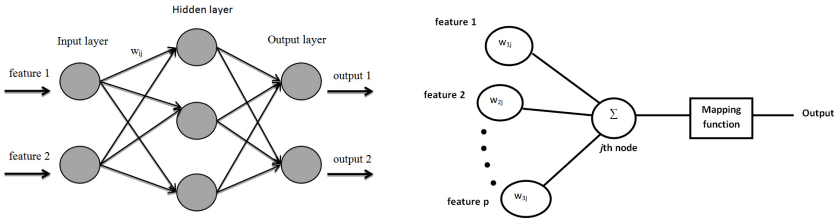
ANN is generally referred to as a mathematical model of the brain activities. An ANN is composed of a number of interconnected processing elements called as nodes that are similar to brain neurons. These nodes are joined by weighted interconnections that are analogous to synapses in the human brain and finally the output paths resemble the axons of the brain. Thus as discussed in Zhang (2000) for the problem of classification, ANN can be defined as a mapping function $F : R^p \rightarrow R^k$ which maps feature space to the class space. In other words the mapping function F takes p-dimensional input data vectors and after mathematical processing returns a k-dimensional vectored output representing the classes of the respective inputs. Supervised classification in an ANN classifier is administered through exposure to a known set of input and corresponding output data i.e. training data. The training algorithm trains the network by adjusting the interconnection weights between the neurons through an iterative procedure such that the overall error is minimized and then this trained network is used to determine the classification of unknown set of data.

Among the five fundamental neural network architectures namely Multilayer perceptron with back-error propagation, the self-organized feature map (SOM), counter-propagation networks, Hopfield networks, and ART systems, the multilayer perceptron with back-propagation considered here for classification is the most widely used supervised ANN architecture design (Tso and Mather (2009); Zhang (2000)).

2.3.1.1 Back-propagation multilayer perceptron (MLP) neural network

A *multilayer perceptron* (MLP) network (Rumelhart et al., 1986) can consist of multiple layers which can be generally categorized into three basic types, first is the input layer, whose nodes take the elements of the external feature vector as inputs, the second type of layer is the hidden layer (which can be more than

one) and the third is the output layer in which the number of nodes is equal to the number of classes in the classification problem. These three types of layers are completely connected to each other with weighted interconnections between the processing elements i.e. nodes of consecutive layers but no connection between the nodes of the same layer. A simplest three layer MLP is shown in Figure 2.1. Each node has its own mathematical function or activation function that accepts input from previous layer and produces output for next layer. The value held by each node is called its activity (a_i). The MLP network is designed with a non-linear activation function in hidden layer and hence aid in non linear mapping between input and output vectors.



(a) 3 layer MLP (b) forward propagation

Figure 2.1: A three layer multilayer perceptron network and the typical working of a processing node in forward propagation.

For training or learning the interconnection weights (w_{ij}) between the layers and the activities of the nodes the back-propagation algorithm is used which consists of forward as well as backward propagation. During forward propagation input signals are supplied to the network through the input layer and the updated activities of the nodes using the interconnection weights, as shown in Figure 2.1 are passed on from layer to layer starting from the input layer to the output layer i.e. from the leftmost layer to the rightmost layer. Formally the input that a single node say j receives is calculated as the weighted sum of the activities of the sending nodes in the preceeding layer defined as

$$x_j = \sum_i a_i w_{ji} \tag{2.1}$$

where, a_i is the activity of the ith node and w_{ji} is the weight of the connection from the ith node to the jth node. And the output from jth node say o_j to the nodes in the next consecutive layer is calculated by converting the input in equation (2.1) using a mapping function, sigmoid mapping function being the common choice. This transfer of updated signal continues from one layer to

another until the output layer is reached. After which the error between the network output and the desired output is computed, which is usually calculated using the least squared error criterion defined as

$$E\left(w\right) = \frac{1}{2}\sum_{j,k}\left(a_{jk} - o_{jk}\right) \tag{2.2}$$

where, w is a set of weights in network, a_{jk} is the jth neurone in the output layer obtained from the kth training sample and o_{jk} is the target output at neurone j in the output layer for the kth training sample (Tso and Mather, 2009). The error $E(w)$ is then back-propagated through the network and the interconnection weights (w_{ji}) are updated according to the generalized delta rule described in Rumelhart et al. (1986) and given in equation below

$$\Delta w_{ji} = \eta\delta_j o_i \tag{2.3}$$

where, η is the learning rate parameter, o_i is the output computed by the ith node, $\delta_j = o_j(1 - o_j)(t_j - o_j)$ with t_j as the target or the desired output for the jth node is the rate of change of error for the output node and $\delta_j = o_j(1 - o_j)\sum_k \delta_k w_{kj}$ for the intermediate node. This process of forward propagation of signals and back propagation of errors is repeated for training samples until the error is minimized or reaches the desired threshold. In this study we used MATLAB's *neural network toolbox* for training artificial neural networks for simulated as well as real datasets (Matlab, 2013a).

2.3.2 Support vector machines (SVMs)

Support vector machines (SVMs) form a group of one of the most recent and theoretically robust machine learning algorithms which were initially developed to overcome the issues of overfitting in machine learning (Vapnik, 1979) but later gained proper elaborative mathematical formulation (Vapnik and Vapnik, 1998). An excellent insight into the working and theoretical development of SVMs apart from what will be discussed in this chapter can be found in Burges (1998). Contrary to the distribution based approach of the traditional MLC and decision boundary-forming logic of ANNs SVMs are aimed at locating an optimal separating hyper-plane between the two data classes in the multidimensional feature space using some optimization algorithms. Under supervised learning, SVMs use training datasets to locate optimal boundaries or hyper-planes between classes and the unseen test datasets are used to ver-

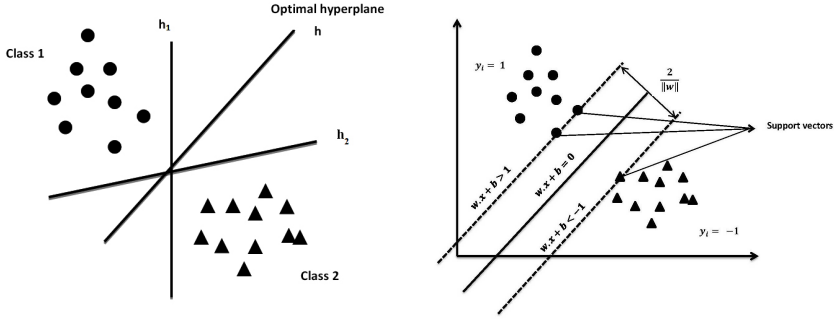
ify their generalizing ability of minimizing the confusion between classes with these optimal boundaries, (Huang et al. (2002); Mountrakis et al. (2011)). SVMs were initially developed as linear binary classifiers allocating labels +1 and -1 to the two classes but were later modified as more versatile classifiers for classifying multiclass data using one-against-one (Knerr et al., 1990) and one-against-others (Vapnik and Vapnik, 1998) techniques. Hence, SVMs can be efficiently applied to multiclass classsification problems also.

For a classification problem involving two p-dimensional data classes, there may be $p-1$ separating hyper-planes but SVMs aim at finding that single optimal hyperplane which minimizes the structural risk by maximizing the distance between the plane and the closest data instances lying on either side of the plane. For example as shown in Figure 2.2, there are 3 separating hyper-planes h_1, h_2 and h but only the hyperplane h fulfills the requirement of minimizing the structural risk i.e. hyperplane h separates the two classes by maiximum margin (Tso and Mather, 2009). The points that constrain the width of the margin between the separating plane and the data instance on either side are called as support vectors and these are generally very less in number.

SVMs have been established as efficient learning algorithms theoretically as well as empirically across various research domains over the years. These structural risk minimizing algorithms have been successfully implemented for varied decision making and classification problems ranging from pattern recognition (Burges, 1998), regression (Hong and Hwang, 2003), clustering (Ben-Hur et al. (2002); Kriegel et al. (2004)), handwriting recognition (Schölkopf et al., 1997), optical character recognition (Joachims, 1998) to remote sensing (Gualtieri and Comp (1998); Huang et al. (2002); Melgani and Bruzzone (2004); Kavzoglu and Kolkesen (2009); Mountrakis et al. (2011)).

2.3.2.1 Theoretical development of SVM

Depending upon the type of separability between the training data classes, SVM algorithms can be divided into two categories. The first one corresponds to the theoretically lesser complex or to the original form of SVM and is used when the training data classes are linearly separable and the other one based on non-linear kernel functions comes in to the picture when the data is found to be linearly inseparable. Both of these are briefly discussed here.



(a) Possible separating linear hyperplanes (b) Support vectors in a linear SVM

Figure 2.2: Linear separating hyperplanes for completely separable classes.

1. *Linearly separable case*: The simplest way of training an SVM is by using linearly separating cases. If we assume p-dimensional linearly separable training datasets represented as $\{\mathbf{x}_i, y_i\}, i = 1, \ldots, n, y_i \in \{1, -1\}, \mathbf{x}_i \in \mathbf{R}^p$, where $\mathbf{x}_i$ represents the p-dimensional set of training vectors and y_i represent the labels of the corresponding classes which is coded as $+1$ for class 1 and -1 for class 2, then the optimum separating hyperplane between the two classes in binary classification problem can be defined as

$$\mathbf{w}^T\mathbf{x} + b = 0 \tag{2.4}$$

where, w is a vector perpendicular to the linear hyperplane and b is the bias representing the offset of the discriminating hyper-plane from the origin.

For linearly separable cases, the hyperplane defined in equation (2.4) is found in terms of two parallel separating hyperplanes, one for each class which are expressed as

$$\mathbf{w}^T\mathbf{x}_i + b \geq +1, \quad \forall \;\; y_i = +1 \tag{2.5}$$

$$\mathbf{w}^T\mathbf{x}_i + b \leq -1, \quad \forall \;\; y_i = -1 \tag{2.6}$$

which can be combined in a single equation as,

$$y_i \left(\mathbf{w}^T\mathbf{x}_i + b\right) - 1 \geq 0. \tag{2.7}$$

These two hyper-planes in equations (2.5) and (2.6) are selected so as not to include any data point in between them while maximizing the distance between the two classes as shown in Figure 2.2 . The training points which lie on these two separating parallel hyper-planes (shaded ones in Figure 2.2) are called *support vectors* (Mathur and Foody, 2008) and have a key role in the establishment of the optimal hyper-plane as they constrain the margin between the training data instances of a class and the separating hyper-plane. The margin between the two parallel hyper-planes described by equations (2.5) and (2.6) is $(2/ \parallel w \parallel)$ and is the distance between the closest points in the two classes, where $\parallel w \parallel$, is the Euclidean norm of w. Thus the optimum separating hyper-plane can be found by minimizing the squared norm, $\parallel w \parallel^2$, of the separating hyper-plane and consequently the problem of locating the optimum hyper-plane reduces to the optimization problem of minimizing the objective function

$$\left[\frac{1}{2} \parallel \mathbf{w} \parallel^2\right] \tag{2.8}$$

subject to

$$y_i(\mathbf{w}^T x_i + b) \geq 1 \tag{2.9}$$

$$y_i \in \{+1, -1\}. \tag{2.10}$$

2. *Linearly-inseparable case*: Mostly the real data encountered in various classification fields is much more complex in nature and is usually found to be linearly inseparable. In such cases it is hard to locate the set of hyper-planes satisfying equations (2.5) and (2.6) which optimally separates the data classes. There may be two situations of inseparability, the first one is that of partial separability. In such a situation when the classes are not completely separable shown in Figure 2.4 by a linear hyperplane, the concept of *soft margin* (Veropoulos et al., 1999) which allows some misclassification contrary to the *hard margin* approach adopted in linearly separable cases, can be used. This method relaxes the constraints in equation (2.10) by introducing a slack variable $\xi_i, i = 1, \ldots, n$, which is proportion to some measure of misclassification cost and indicates the distance of misclassified points from the optimal hyperplane (Oommen et al., 2008). Thus the optimization problem de-

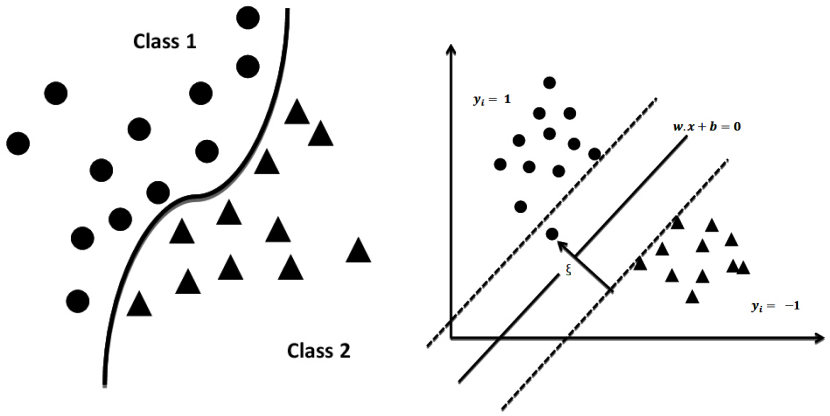
scribed above modifies to

$$min(\frac{1}{2} \parallel \mathbf{w} \parallel^2 + C \sum_{i=1}^{n} \xi_i) \tag{2.11}$$

subject to

$$y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad \forall i \tag{2.12}$$

where, C is the regularization parameter or the penalty parameter which regularizes the balance between the two parallely acting criteria of margin maximization and error minimization in SVM. The larger is the value of the penalty parameter, higher is the penalty it associates to the misclassified samples (Melgani and Bruzzone, 2004). And a linear separating boundary can still be located between the classes as shown in Figure 2.4



(a) Partially separable data (b) Optimal hyperplane

Figure 2.3: Linear separating hyperplanes for partially separable classes using soft margin concept .

The second type of inseparability is that of complete inseparability between the training samples and the approach adapted to resolve this issue is that of mapping where a non-linear mapping function say Φ is used to map the original training data classes into higher dimensional feature space (Aizerman et al., 1964) where they can be linearly separated. And linear optimal hyper-plane is then fitted between the classes in the new higher dimensional transformed feature space as depicted in Figure 2.5. An appropriately chosen transformed feature space of sufficient dimensionality is found to be capable of discriminating between the data classes (Kotsiantis et al., 2007) as shown in Figure 2.4. The

linear optimal hyper-plane in the transformed space corresponds to the non-linear one in the original feature space. The classification decision function for non-linear SVM is defined as,

$$f(x) = sign\Big(\sum_{i}^{sv} \alpha_i y_i \Phi(\mathbf{x}).\Phi(\mathbf{x}_i) + b\Big). \tag{2.13}$$

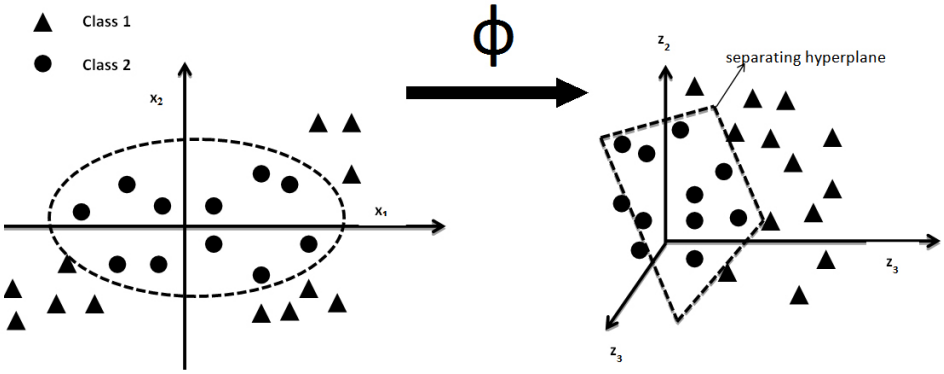


Figure 2.4: Separating hyperplane for inseparable classes using higher dimensional feture space.

where, $\Phi : \mathbf{R}^p \rightarrow H$ is the mapping function from p-dimensional input space to the higher dimensional transformed feature space, α_i are the Langrage multipliers, and sv is the number of support vectors. The magnitude of α_i is determined by the penalty parameter C. The computaional burden of $(\Phi(\mathbf{x}) \times \Phi(\mathbf{x}_i))$ for mapping input data $\mathbf{x}$ can be quite expensive (Tso and Mather, 2009). As a solution, Vladimir and Vapnik (1995) propose the computationally more efficient *kernel function* approach to map the input data into the transformed feature space. A kernel function is denoted as $K(\mathbf{x}, y)$ such that $K(\mathbf{x}, y) = \Phi(\mathbf{x}) \times \Phi(y)$ the decision function in equation (2.13) modifies to

$$f(x) = sign\Big(\sum_{i}^{sv} \alpha_i y_i K(\mathbf{x}, \mathbf{x}_i\}) + b\Big). \tag{2.14}$$

Choice of the kernel function, used for transforming the feature space plays an important role in determining the performance of the trained SVM as kernel function determines the feature space in which the original data is mapped. Unfortunately, there are no set rules for determining the appropriate kernel function type for a given problem and they have to be determined heuristically

or by rule of thumb (Genton (2002); Tso and Mather (2009)). However, once the user gets success in selecting a legitimate kernel function type after trying a range of potential settings and cross validating them, the associated parameters can be optimally selected by using either the *grid search* (Chang and Lin, 2011) or the *gradient descent* technique introduced in (Chapelle et al., 2002). There are four major types of kernel functions used for training SVM classifiers, the polynomial kernel function, the radial basis function (rbf) and the sigmoid kernel function. Studies suggest that sigmoid kernel usually does not perform ideally for classification problems. Whereas the performance of polynomial kernels and the *rbf* kernel is found to be comparable with *gaussian rbf* kernel usually being the preferable choice (Tso and Mather, 2009). Hence, in the present study *gaussian rbf* has been used for training the SVM classifier and its parameters are learned using the gradient search method. The *gaussian rbf* kernel is defined as,

$$K(\mathbf{x}_i, \mathbf{x}_j) = exp\big(- \parallel \mathbf{x}_i - \mathbf{x}_j \parallel^2 /2\sigma^2\big). \tag{2.15}$$

where, $\mathbf{x}_i, \mathbf{x}_j$ are the feature vectors, σ is the so called free parameter which along with the error penalty parameter C need to be fixed by the user.
The major advantages of SVM classifiers over others is their ability to minimize the misclassification rates for unseen samples originating from fixed but unknown probability distributions, structural risk minimization (SRM) concept based training which always finds a global minimum (Tso and Mather, 2009), higher generalization capabilities as compared to ANNs and lesser efforts required for training the model parameters (Joachims, 1998). Moreover, during learning the number of support vectors selected by SVM to determine the model is usually very small as shown in Figure 2.2 and hence, SVM based classifiers are less affected by the scarcity of training data. This property makes SVM well suited classification choice when the ratio of training data instances to the number of features is very large as is encountered in microarray datasets. In general, SVMs have been found to be performing better than the traditional parametric approaches (Huang et al. (2002); Kavzoglu and Kolkesen (2009); Tso and Mather (2009); Otukei and Blaschke (2010); Khondoker et al. (2013)) in terms of higher classification accuracies. But the extent of success of SVMs in discriminating between the classes depends largely upon how well they are trained in terms of the method used to generate SVM model, choice of kernel parameters and the choice of parameters for the chosen kernel

as well (Huang et al., 2002). Also, SVM based classifiers are supposed to be sensitive to the outliers (Shao and S., 2012) and hence, a critical empirical analysis of their performance for classifying skewed datasets as is attempted in the present chapter will further help the practitioners in selecting the most appropriate classifier for dealing with the skewed datasets.

2.3.3 Random forests (RFs)

Before formally describing the random forest classifiers, some underlying methods and concepts which collectively form an integral part in the development of RFs are needed to be defined.

- *Decision trees (DTs)*: Decision tree is a non-parametric classification technique that performs classification using the hierarchical splitting approach where labeling of an unknown pattern is done using a sequence of if-then decisions rules (Tso and Mather, 2009) and hence provides more comprehensive understanding of the relationships between the input data and the output labels as compared to the more complex ANNs. The typical structure of a decision tree classifier is shown in Figure 2.5 (Strickland, 2015) which basically consists of a *root* node which contains the whole of the input data, internal nodes or the *leaf nodes* at each of which splitting of the data is performed depending upon the algorithm used for growing the tree, and the terminal nodes which represent the final outcome or the corresponding label of the input feature vector. The decision tree shown in Figure 2.5 takes the *petal width* of a flower from Fisher's iris data (Fisher, 1936) as input to classify it as belonging to one of the three species which are *setosa, versicolor* and *virginica*.

  Classification and Regression trees (CARTs) are one of the most widely and frequently used DTs in the classification field. CART builds a tree by recursive binary partitioning of the input data into the nodes that are increasingly more homogeneous with respect to the class variable (Cutler et al., 2007a). At each classification step, selection is made for a node, predictor feature and a cut-off using optimization which result in the most homogeneous subgroup of the data as measured by Gini index (Breiman et al., 1984). The splitting process is continued until all resulting sub divisions are pure or until the further partitioning doesnt reduce the Gini index. Such a tree is called a fully grown tree with terminal nodes as

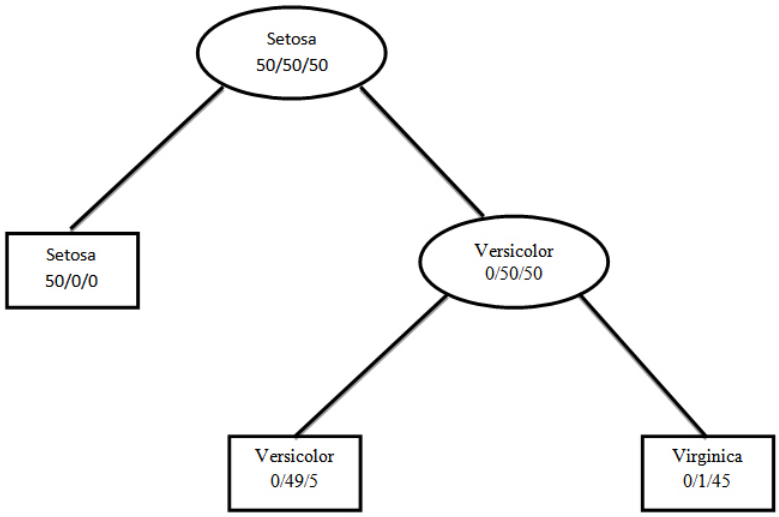


Figure 2.5: A simple decision tree for the classification of iris data.

the final sub-groups. CARTs are found to over fit the training datasets and hence, pruning techniques such as cross validation are used to prune the over fitted tree to an optimal one (Tso and Mather, 2009). Besides being distribution free methods, decision trees are well adapted to deal with heterogeneity and the noisy data, and also these are found to be effective at choosing from large number of feature variables. Hence, DT classifiers can be very efficient in the classification of microarray datasets and hyperspectral image data. Despite all these advantages DTs performance can be affected by small changes in the data (Prasad et al., 2006). And to maintain the stability of the trees, advanced ensemble learning techniques based on bagging have been proposed (Breiman, 1996a).

- *Ensemble learning/ Boosting/Bagging:* In the recent years ensemble learning that generates many classifiers and aggregate their results for making final decisions has gained a lot of research interest. The ensemble learning methods have been proven to always give better classification accuracies theoretically as well as empirically than an individual classifier (Krogh et al., 1995). These machine learning methods have successfully been applied and are shown to give more accurate results as compared to the single classifiers for various classification problems (Opitz and Maclin (1999); Kosorok et al. (2007)). The ensemble of classifiers is generated using re-sampling techniques. Bagging (Breiman, 1996a) and boosting

(Schapire, 2003) are the two well-known re-sampling techniques that are often used to generate the ensembles. In boosting, successive trees give extra weights to points incorrectly predicted by earlier predictors and in the end a weighted vote is taken for prediction. Bagging is a re-sampling technique which works on the concept of aggregated bootstrap samples. In bagging of decision trees, successive m independent fully grown trees are generated using N bootstrap samples of the training dataset of size, say N, each of the m fully grown trees without pruning cast a vote in favour of one of the possible k classes and in the end a simple majority vote decides the final prediction of the input feature vector. The basic idea behind using bagging in classification trees is to avoid the situation when the output error of a single classification tree could be due to the specific choice of the training sample. And hence, if independent classification trees are grown without pruning with several similar bootstrap samples generated from the original data, the output variance in the error is reduced (Breiman, 1996a).

- *Random forest (RF) classifiers:* RF classifiers originally developed by Breiman (2001) correspond to the relatively latest classification algorithms which attracted wide scale interests of the researchers in a relatively smaller duration since their development. Random forest algorithms belong to the class of *ensemble learning* algorithms which have been shown to be effectively useful in although not numerous due to its most recent discovery but still in a considerable number of significant researches. As its name suggests, an RF classifiers architecture is based on the concept of generating a forest or an ensemble of a large number of bagged classification trees which are grown on random subset of input vectors and splitting nodes on a random subset of features (Prinzie and den Poel, 2008). The main difference between the construction of trees in RFs and in CARTs is that in CART each node is split using the best split among all variables, while in a random forest, each node is split using the best variable among a subset of predictors randomly chosen at that node. This strategy increases the randomness in bagging the trees and hence turns out to perform pretty well as compared to the other advanced machine learning algorithms like ANNs and SVMs (Breiman, 2001). Figure 2.6 depicts the working of RF with a simple case of ensemble learning using decision trees for a two classs problem. Here each of

the three fully grown decision trees cast a vote in favour of one of the two classes, and the final class prediction has been decided by a majority vote.

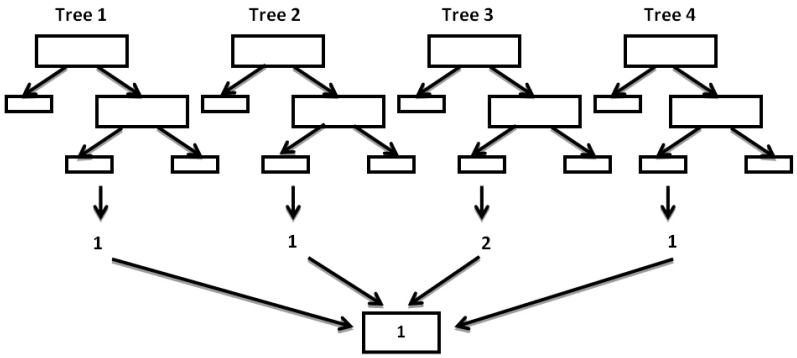


Figure 2.6: Working of a random forest.

Random Forests techniques have been successfully implemented in a number of researches across a varied range of fields but still the researches investigating the robustness of RF classifiers for various types of simulated and real datasets is still relatively very less as compared to other classifiers. Many studies, (Pal (2005); Chan and Paelinckx (2008); Waske and Braun (2009); Martinuzzi et al. (2009); Ghimire et al. (2010); Lawrence et al. (2006); Latifi et al. (2010)) have studied the robustness of RF classifiers for the land cover classification of multispectral and hyperspectral satellite sensor imagery. In ecology, Cutler et al. (2007a) justified the better performance of RF for classification of invasive plant species, Liaw and Wiener (2002) tested the performance of RF techniques for various benchmark datasets and found their performance to be favourable comparable with that of SVM, (Prasad et al., 2006) found RF and Bagged Tree (BT) classifiers as robust tools for predictive vegetation mapping and suggested their inclusion in ecological toolboxes. Prinzie and den Poel (2008); Gall et al. (2012); Kulkarni and Sinha (2013) and Kulkarni and Sinha (2014) have employed RF techniques for varied application fields and lauded these non-parametric classifiers for their efficient at par performance in terms of improved classification accuracy with other ensemble techniques like *bagging* and *boosting* as well as with ANN and SVM. The application field which has explored the RF's classification capabilities the most is that of gene selection and microarray based cancer classification. Huang et al. (2005); Diaz-Uriarte and de Andres (2006); Statnikov et al. (2008); Khondoker et al. (2013) and Zakariah (2014) are a few of the publications which employed RF classifiers in

microarray data classification.

The RF classifiers possess various attractive advantages over other classifiers. They do not need extensive parameter training like SVM and ANN and are required to be provided with only two parameter values i.e. the number of trees to be grown and the number of predictors to be considered for best split at each node. Moreover, the parameters do not need much fine-tuning and often the default parameter values give desirable results. Out-of-bag samples (Breiman, 1996b) at each boostrapping step can be used to calculate an unbiased error rate and variable importance which eliminates the need for a separate test set for cross validation (Breiman, 2001). RF classifier performs embedded feature selection and is found to be relatively insensitive to large number of irrelevant features, and hence spares the user of some pre-processing load of feature selection. Classification by random forest tehniques results in very limited generalization error due to the construction of a large number of trees and hence leaves no or very little scope for overfitting. Its random predictor selection strategy diminishes correlation among the unpruned trees and keeps the bias low (Prasad et al., 2006).

In contrast to all these appealing advantages, RFs do not have many disadvantages except that they unable the examination of individual trees separately and are relatively slow as compared to SVMs and parametric MLC due to the construction of a large number of trees. Apart from all the above discussed advantages, RFs are found to be relatively more robust to outliers and to noise and this characteristic of RF classifiers might prove to be beneficial for the classification of highly skewed datasets. Recognizing the caliber of RF classifiers in efficiently classifying complex data types, the present work investigates their robustness for classifying highly skewed datasets. Hence, the investigations carried out in the present work are expected to contribute some more facts to the field of application of RF classifiers.

2.3.4 Accuracy assessment

The quality of a classifier is judged by its predictive accuracy on unknown samples or in terms of misclassification probabilities. There may be many ways for calculating the predictive accuracy of a classifier and the choice of the most appropriate classification assessment rule must be made on the basis of some objective considerations suggested in Fielding and Bell (1997). The simplest descriptive technique for error assessment is the calculation of overall

misclassification rates. However, depending upon the objective of a classification problem, one should give a thought to choosing the most efficient one amongst other assessment measures like User's / Producer's accuracy, ROC Curve, Kappa coefficient etc. (Richards and Richards, 2008). For example, in case of imbalanced datasets or when the problem requires the consideration of a cost matrix , calculations of User's / Producer's accuracy which takes into account the effect of prevalence of the classes gives a more justified picture of the actual misclassifications. The simulation study reported here is designed with balanced datasets sans any cost constrained with an aim to compare the overall misclassification proportions produced by the different classifiers. The measures of overall misclassification rates are expected to provide acceptable results in context of the present study and hence have been used for comparing the predictive accuracies of the classifiers. To assess the performances of the various classifiers uesd in this study, the Apparent Error (APE) rate which is calculated by taking the expectation of the total misclassification proportions of training data over repeated samples as well as the Actual Error (AE) rates which tend to be better estimates of misclassification probabilities and are calulated by taking the expectation of misclassification proportions of test data over repeatedly trained classifiers were obtained for all the simulated datasets. Apart from overall misclassification rates, measures of chance agreement were also calculated using Gwet's AC1 statistic in order to assess the reliability of the classifers.

2.4 Numerical Experiments and Results

2.4.1 Simulation and data generation

The main aim of this chapter is to study the robustness of various machine learning algorithms for classifying skewed datasets. With a purpose to zero in the optimal non-parametric machine learning algorithm in terms of the lesser misclassification error rates for classification of skewed data, an extensive simulation study has been carried out in this section with a variety of simulation settings generating moderately as well as highly skewed datasets. There may be a number of factors or data characteristics, such as variability, training data size, separability between the groups, feature set size etc., which can affect a classifiers performance apart from the skewness of the data. Hence, in the present study, the training as well as the test datasets are generated

for diverse combinations of such data characteristics in order to study their individual as well as interactive effects on the classifiers performance. For simulating the skewed datasets for a varied range of skewness, we follow the methodology that was used in Clarke et al. (1979). We generated training datasets from multivariate normal populations with varied configurations for various combinations of the factors that could have affected the classification performances. The first population was simulated from standard multivariate normal distribution while several configurations of the population parameters, which are population mean and population variance-covariance matrix for a multivariate normal population, were considered for simulating other multivaraite normal populations. The other parameters that were varied at each step of the simulation in order to make the simulations more diverse are skewness parameter (δ), dimensionality of data (p) and size of the training dataset (n). We conducted simulations for bi-variate as well as ten-varite datasets in the present study.

An account of the configurations of parameters which were considered for simulations is given in Table 2.1. After simulating the multivariate normal populations using the parameter values given in Table 2.1, the transformations in equation (2.16) were used to generate multivariate skewed data from the simulated multivariate normal data. Each of the three classification algorithms namely ANN, SVM and RF was trained using the simulated training datasets. A separately generated skewed index sample of size 800 was classified by the trained classifier and the resulting misclassification error rates were calculated. This process of training and validating a classifier was repeated over 30 replications for training and index datasets simulated for each of the parameter combinations given in Table 2.1 and the observed misclassification error rates were averaged over all the replications to get an unbiased estimate of the misclassification error rates. Apart from the actual error (AE) rates which were calculated for the index sample, the apparent error (APE) rates based on the misclassification probabilities of the training samples were also calculated and compared. This computation was repeated for each of the 3 classifiers under investigation in this study and the results are summarized in Table 2.5 and 2.6.

Apart from the parameter combinations that were inherited in the present study from Clarke et al. (1979) for skewed data to make the results in the two studies comparable, we extended our analysis by simulating severely skewed data for $\delta = (0.5 \text{ and } 0.9)$. If $\mathbf{X_i} \sim \mathbf{N_p}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ then transformations for gener-

ating multivariate skewed data $\mathbf{Y_i}$ are given as

$$\mathbf{Y_i} = \exp\left(\mathbf{X_i}/\delta\right) \tag{2.16}$$

where, δ is used to generate a range of skewness throughout the simulations for simulating multivariate skewed data. The Mardia's multivariate coefficient of skewness (Mardia (1970, 1974)), which gives a measure of the skewness of multivariate data, was calculated for differently skewed index samples over a range of δ and are tabulated in Tables 2.3 and 2.4.

The classification process was carried out in *MATLAB* using the *Neural Network toolbox* (Matlab, 2013a) for ANN, *svmclassify* function for SVM for a two class problem and *TreeBagger* function (Matlab, 2013b) for generating RF. The ANN classifier was trained with back propagated multilayer perceptron algorithm for different settings of the hidden layer sizes and was found to be performing the best for a value of 15. The non-linear SVM classiifer was trained with *gaussian rbf kernel* and the values of the parameters were fixed using grid search method. The number of optimal trees for RF was determined hueristically and it was fixed at 500 over all the simulations. To measure the significance of the levels of agreements produced by the three classifiers, the Gwet's AC1 statistic (AC1) (Gwet, 2002) which is discussed in detail in next chapter, was used. An average measure of the AC1 coefficient values over the 30 replications was calculated for each of the three classifiers.

Number of Variables	$p = (2, 10)$
Mean Vector of second population	$\boldsymbol{\mu_2} = \left(a^{\dagger}, 0, \ldots, 0\right)$ $a = (0, 1, 2)$
Covariance Matrix of second population	$\boldsymbol{\Sigma_2} = \sigma^2 I$ $\sigma^2 = (1.5, 3, 8)$
Skewness Parameter	$\delta = (0.5, 0.9, 2, 5)$
Size of Training sample from each class	$n = (25, 50, 100, 400, 6001000)$

Table 2.1: Parameter combinations for simulations.

2.4.2 Real datasets used for comparison

We also evaluated and compared the performance of ANN, SVM and RF classifiers for classifying positively skewed data on some benchmark real life

[†] $\mathbf{a}$ is the mean of the first variable of second population, which ensures the variation in the separation between the two populations

datasets. Two datasets have been chosen from the field of digital imaging and two from the field of diagnosis. namely the LANDSAT dataset (Bache and Lichman, 2013), the SPOT dataset (), the New Thyroid Dataset and the Indian Liver Patient Database (ILPD) (Bache and Lichman, 2013) which have previously been used in various studies. An account of them is given below.

1. *Dataset 1:*

   The Landsat satellite data can be one of the many sources of information that may be available for a scene. One frame of Landsat MSS imagery consists of four digital images of the same scene in different spectral bands. Two of these are in the visible region (corresponding approximately to green and red regions of the visible spectrum) and two are in the (near) infra-red. Each pixel is an 8-bit binary word, with 0 corresponding to black and 255 to white. The spatial resolution of a pixel is about $80m \times 80m$. Each image contains (2340×3380) such pixels. The Landsat database (Bache and Lichman, 2013) consists of 6435 instances on 6 landuse classes namely the red soil, cotton crop, grey soil, damp grey soil with vegetation stubble and very damp grey soil which are present in a (tiny) subarea of a scene captured by Landsat satellite. The scene consists of (82×100) pixels. Each of the 6435 rows of the data corresponds to a (3×3) square neighbourhood of pixels completely contained within the (82×100) sub-area and a number indicating the classification label of the central pixel. Hence, we have used only the central pixels of each of the (3×3) neighbourhood of pixels ignoring the other pixels. It implies that each row contains the pixel values in the four spectral bands (converted to ASCII) on 9 pixels. After considering only the central pixels for classification the sizes of the training and the test datasets reduce to 4435×4 and (2000×4) respectively where rows correspond to each of the 6435 pixels and columns correspond to their spectral in the four spectral bands. All the 6 classes in this dataset were found to be significantly skewed with values of Mardia's multivariate coefficient of skewness at $2.52, 6.12, 2.12, 1.414, 4$ and 2.046 respectively.

2. *Dataset 2:*

   The SPOT dataset (Glasbey, 1988) is a (10000×2) array of instances

on 8 landuse categories. It contains spectral values of 100000 pixels in two spectral bands. For this dataset, 50 random observations from each of the 8 classes were placed into the test dataset and the remaining were contained in the training dataset. In this way the training dataset is a (400×2) array of 400 instances. Class $1, 4, 6$ and 8 were found to be significantly positively skewed in the training dataset with values of the Mardia's multivariate coefficient of skewness as $3.14, 3.03, 2.11$ and 2.17 respectively.

3. *Dataset 3:*

   The New Thyroid Dataset is a (215×5) data array containing the measurements of 5 attributes (which are 5 Lab tests) on each of the 215 patients in order to predict a patient's thyroid state as normal, hypothyroidism or hyperthyroidism. On the basis of the lab tests, out of 215 instances in the dataset 150 of them were found to be in the normal thyroid range, 35 in the hypothyroid and 30 in the range of hyperthyroidism. All 3 of the classes in the dataset tested positive for significant multivariate skewness with the coefficient of multivariate skewness values $5.14, 6.69$ and 11.753 respectively.

4. *Dataset 4:*

   The ILPD is a (583×10) array containing a total of 583 patient records on 10 attributes. Out of 583 cases, 416 are attributed to the liver patient category and the remaining 167 to normal liver functioning patients category. For this dataset too the multivariate skewness coefficient for the two classes were found to be significant at values 543.38 and 97.96 respectively.

Dataset 1 has already been given with separate training and test dataset at Bache and Lichman (2013) and dataset 2 contains sufficient number of instance in each of the classes which can be easily divided into training as well as test datasets separately. Hence, the actual error rates (AER) for these two datasets were evaluated by training the classifier using the training dataset and validating it with the separate test dataset. While the actual misclassification error rates for datasets 3 and 4 were obtained using Lachenbruch's leave one out method (Lachenbruch, 1975) of cross-validation. An initial uni-variate plotting of the four datasets hinted at the non-symmetric nature and the presence of

positive skewness in the data. ANN, SVM and RF classifiers were used to classify each of the four datasets independently and the APER, AER and learning times of each of the three classifiers were calculated and are reported in Table 2.6.

2.4.3 Results

2.4.3.1 Results on simulated data

An extensive simulation study was performed in this chapter with an objective to compare the classification performance in terms of misclassification error rates of the now becoming popular non-parametric classifiers based on ANN, SVM and Random forest techniques while handling positively skewed datasets. The Actual error rates (AERs) of the simulated index samples and the Apparent error rates (APERs) over the 30 replications of the simulated training datasets for the three classifiers ANN, SVM and RF are tabulated in Tables 2.5 and 2.6 for ($\delta = .5$) and ($\delta = .9$) respectively. And the plots of the AER against the values of the various data characteristics are shown in Figures 2.7, 2.8 and 2.9. Plots in Figure 2.7 correspond to ($\delta = .5, p = 2$) and depict the effect of training sample size (n) on the actual error rates of the three classifiers. Plots in Figures 2.8 and 2.9 correspond to ($\delta = .5, p = (2, 10), n = 25$) and ($\delta = .5, p = (2, 10), n = 100$) respectively and depicts the effect of the data varaibility on the AERs of the three classifiers. Since the trends of the AERs were found to be same for ($\delta = .5$) and ($\delta = .9$), hence the plots have been shown only for ($\delta = .9$). The tendency of the apparent error rates to underestimate the misclassification probabilities is clearly evident from the Tables 2.5 and 2.6. The average agreement measure in terms of AC1 coefficient values for the index samples calculated over all the 30 replications are tabulated in Table 2.3. Also the following findings were observed for various levels of the data characteristics.

- *Effect of delta:* Tables 2.3 and 2.4 depict the tendency of δ and σ to produce variation in the skewness of datasets. The skewness in the datasets increases as value of δ decreases from 5 to .5 and value of σ increases from 1.5 to 8.

- Among the three classifiers the RF classifier was found to be the best performer for all the simulated datasets which vary over a number of data characteristics except for the data simulated with ($a = 0, \sigma = 1.5$) i.e.

when the means of the two populations were same and the variability of the second population (which affects the skewness of the data) was less where SVM outperformed RF by a small margin. While the ANN classifier's performance was found to be worst in terms of the misclassification error rates produced by the three classifiers.

- *Effect of training sample size:* Training sample size considerations were found to be important as it is evident from Tables 2.5 and 2.6 that the misclassification error rates depicted an inverse proportionality to the training sample size for all the three classifiers under study. It can be observed from the plots in Figure 2.6 that for RF classifiers the error rates continuously decrease as the training sample sizes are increased from 25 to 50. SVMs depict same trends for moderately skewed datsets i.e. for ($\sigma = 1.5$) but rather showcased the tendency of producing larger error rates for larger sample sizes as the variability of the datasets increase with σ. It can be observed from these plots that for all the three classifiers the most considerable decrease in the error rates was observed as the sample sizes are increased from 25 to 50 and a very small improvement afterwards. Hence, we have plotted the error rates against other data characteristics only for ($n = 25$) and ($n = 100$). Across all the variations in sample sizes considered here, RF emerged as a clear winner in terms of producing smaller misclassification error rates.

- *Effect of Skewness:* For lower ranges of skewness in datasets i.e. for datasets generated with $\delta =$ (2 and 5), (results not shown here) SVM and RF classifiers performance was found comparably similar but as the value of δ drops below 1, i.e. for $\delta =$ (.5 and .9) the skewness of the marginal distribution of second population increases, and hence the gap between the misclassification probabilities obtained from the three classifiers starts increasing. It can be seen from Table 2.5 that in all the above discussed situations of severe positive skewness in the data, performance of the RF classifier was fairly better than that of the SVM and ANN classifiers. Also, as the levels of skewness were increased by increasing the values on the diagonal of covariance matrix of second population, RF outplayed the other two classifiers with a clean majority. Also all the classifiers showed the obvious trends of improved performance with the increase in the distance between the classes which was varied be-

tween the two populations in the simulated datasets with the value of $a^{\dagger}$. Apart from the comparartive performance, indvidually all the three classifers depicted a tendency of deteriorating performance at higher levels of skewness i.e for larger δ.

- *Chance agreement measures:* The values of average Gwet's AC1 coefficients for RF reported in Table 2.7 were observed to be lying in the range $(0.5, 0.7)$ implying fair to moderate levels of agreement with AC1 measure improving over the separabiltiy between the two classes. For RF classifiers the average values of the AC1 measure lied in the range $(.5, .8)$ which reports a fair to good level of agreement. The AC1 measure for RF classifiers improved with the increasing separability between the two classes as well as with the increased skewness of the datasets. The level of agreement for ANN classifiers was not found to be improving at all with a constant value of average AC1 measure at .5.

- *Effect of dimensionality:* On an average the performance of SVM and ANN classifiers was found to be deteriorating when the number of featutres in the datasets was increased from 2 to 10. Only the RF classifier stood the test of dimensionality and its performance was found to be improved for the increased dimensionality of the datasets.

- The comaparatively poor performance of SVM relative to the RF classifier in the present study might be accounted to the sensitivity of SVM classifier to the presence of outliers in data (Shao and S., 2012) or to the possible discrepancy in learning the parameters efficiently, despite all efforts. And any adjustments in the parameters, using other parameter selection methods might lead to a different set of results. This observation itself reports the dire need of proper training of an SVM classifier, which might not be possible for a non-expert user.

- The extremely good performance of RF classifier may be accounted to the ability of RF's in handling highly skewed variables and the outliers efficiently (Shi and Horvath, 2012).

It was observed from the results of the simulation study that the RF classifier performed fairly better than the classifiers based on SVM and ANN for

$^{\dagger}a$ is the mean of the first variable of second population, which ensures the variation in the separation between the two populations

heavily skewed simulated data (i.e. for $\delta = (.5 \text{ and} .9)$) over all the other data characteristics that were considered in this study. Although RF classifier performed comparably well for the skewed datasets for all the combinations of the different levels of various data characteristics but its tendency of overfitting the training data and the very large amount of computational time, it takes as compared to the MLC, makes it a not so attractive and feasible option for classification of very large datasets.

2.4.3.2 Results on real datasets

Athough the real datasets considered in the present study were found significantly skewed by Mardia's test for some of the classes but none of the classes in any of the dataset was found to be highly skewed which is the main assumption in this study. Still all the four datasets were classified using ANN (MLP-BP), SVM (with gaussian rbf kernel) and RF classification algorithm and the misclassification errors for them are reported in Table 2.2. The AERs for dataset 1 and dataset 2 were calculated using a separate index sample while for dataset 3 and 4 leave-one-out cross-validation errors were calculated due to the limited number of observations present in these two datasets. The values reported in the brackets with the AER denote the optimal parameter values which were used for training each of the three classifiers. For ANN the parameter is the size of the hidden layer, for SVM its the kernel parameter and for RF it is the number of trees used for generating the forest. The following conclusions can be drawn from the results obtained.

- For dataset 1, all the three classifiers performed comparably well.
- ANN and RF performed quite fairly for all the datasets except the dataset 2 which is the SPOT data. The huge class imbalance in SPOT data is a reason for the poor performance of the two classifiers.
- SVM reported maximum classification errors among the three classifiers for all the datasets except the SPOT dataset which exhibits the problem of class imbalance. This observation suggests that among RF, ANN and SVM classifiers, SVM classifier is least affected by the class imbalance of the dataset. The poor performance of SVM on real datasets might be attributed to the inability of SVM to transform non-linearclass boundaries in the original space to the linear ones in a higher dimensional space or to the possible discrepancy in learning the parameters ef

ciently, despite all efforts. This observation itself reports the dire need of proper training of an SVM classifier, which might not be possible for a non-expert user.

	ANN		*SVM*		*RF*	
	APE	AE	APE	AER	APE	AE
Dataset 1	12.41	13.81 (15)	6.95	16.40 (1)	4.13	16.35 (100)
Dataset 2	55	56.08 (15)	32.47	39.90 (2)	23.25	69.86 (50)
Dataset 3	.10	.47 (15)	3.23	9.77 (2)	0	4.65 (50)
Dataset 4	25.77	29.91 (15)	29.71	37.82 (1)	0	29.02 (100)

Table 2.2: Apparaent error rate (APER) and Actual error rate (AER) of ANN, SVM and RF with their respective training parameter values for the real datasets.

2.5 Conclusion

This chapter focussed on the need of specialized treatment of highly skewed datasets. An attempt was made using the simulated datasets to select the most robust non-parametric alternative to the maximum likelihood classifier from a group of three most advanced non-parametric classification algorithms which are support vector machines, artificial neural networks and the random forest for classifying positively skewed datasets. Results of the investigations carried out on simulated data provide empirical evidences that the random forest algorithm is highly robust even to the very large levels of positive skewness in the datasets. In the light of other advantages discussed in this chapter such as lesser learning effort, that random forest classifiers enjoy over its counterparts support vector machines and artificial neural network classifiers, we conclude that random forest classifiers should be preferred over the SVM and ANN while dealing with severely positively skewed data. However, for moderate levels of skewness one can also choose computationally much faster SVM classifier as it was found to be performing comparably well. Moreover, on the basis of the empirical results obtained in this study we keep ANN classifiers at bottom in the list of feasible non-parametric options for classifying highly skewed datasets on account of their poor performance.

Need of considerable learning efforts for training, sensitivity to the outliers and complex computations are some issues of the non-parametric machine algorithms that limit their performance while classifying highly skewed datasets

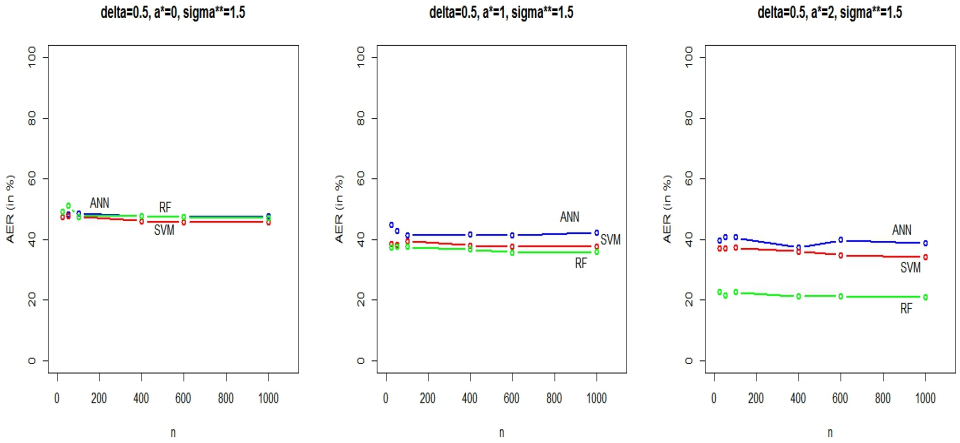
and hence require considerable attention of the researchers. In order to address these issues, next chapter explores the potential of parametric classifiers for efficient handling of highly skewed datasets.

		$\delta = .5$		$\delta = .9$	
		S_k		S_k	
σ^2	a	Pop.1	Pop. 2	Pop.1	Pop. 2
1.5	0	155.4415	478.93	45.59	158.96
	1	155.2869	288.42	30.5929	64.5322
	2	191.03	400.11	41.39	134.57
3	0	118.68	607.63	32.84	333.09
	1	162.19	453.63	39.63	217.64
	2	72.83	540.11	17.49	217.38
8	0	181.87	516.40	34.93	464.84
	1	430.45	666.76	121.07	430.83
	2	141.15	707.89	42.68	529.44

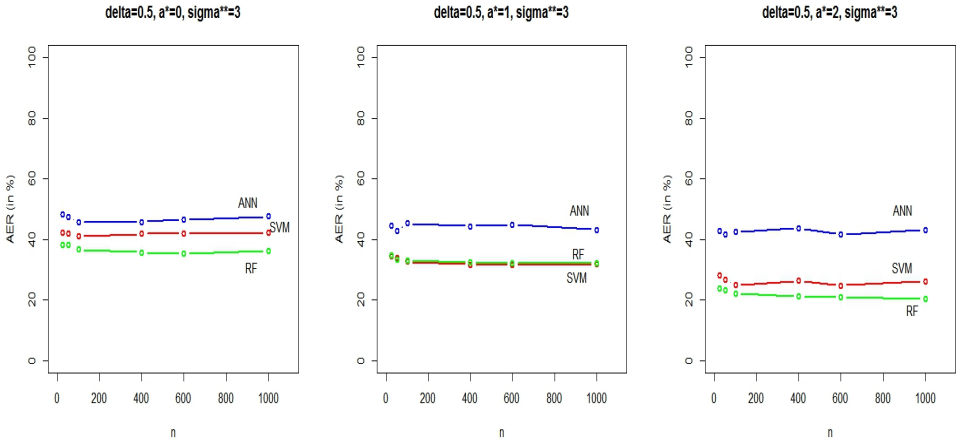
Table 2.3: Classwise Mardia's multivariate coefficient of skewness (S_k) for simulated bivariate index samples .

		$\delta = .5$		$\delta = .9$	
		S_k		S_k	
σ^2	a	Pop.1	Pop. 2	Pop.1	Pop. 2
1.5	0	1150.99	1144.94	501.94	334.33
	1	1596.44	1450.10	355.19	525.89
	2	1207.59	1456.29	269.96	408.83
3	0	925.04	2213.18	239.63	1405.56
	1	842.27	1976.96	194.23	970.24
	2	832.38	2377.19	188.04	1553.15
8	0	1566.87	2157.29	533.11	1444.29
	1	1086.42	3044.94	249.96	2286.89
	2	1260.11	2462.62	343.73	1699.16

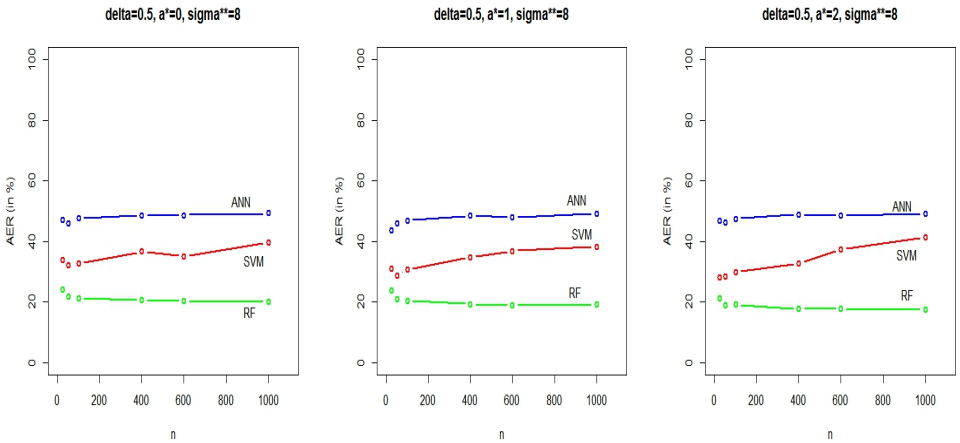
Table 2.4: Classwise Mardia's multivariate coefficient of skewness (S_k) for simulated ten variate index samples .



(a) $p = 2$ (b) $p = 2$ (c) $p = 2$

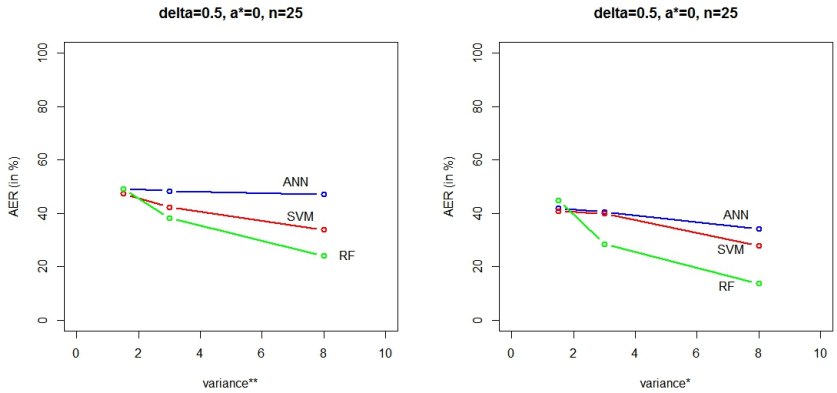


(d) $p = 2$ (e) $p = 2$ (f) $p = 2$



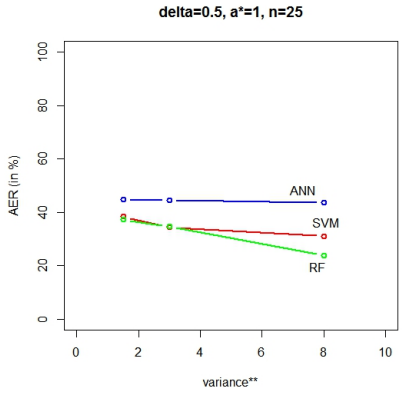
(g) $p = 2$ (h) $p = 2$ (i) $p = 2$

Figure 2.7: Plots of expected actual error rates of SVM, RF and ANN over simulated index sample for $\delta = .5$ depicting the effect of training sample size on error rates .

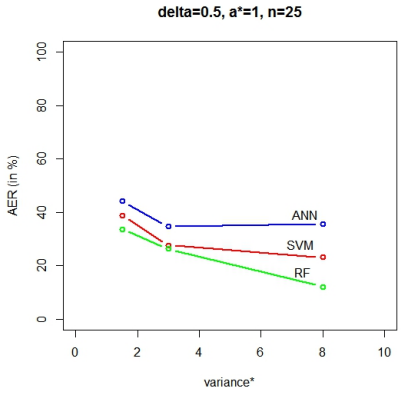


(a) $p = 2$

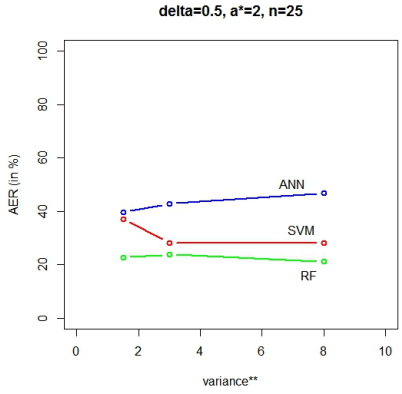
(b) $p = 10$



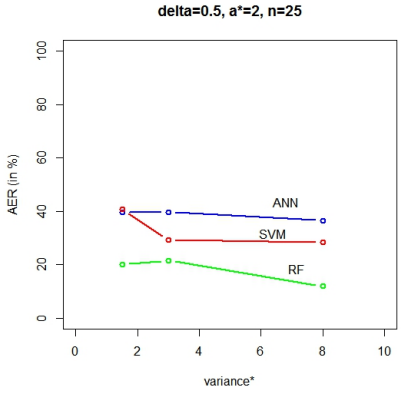
(c) $p = 2$



(d) $p = 10$

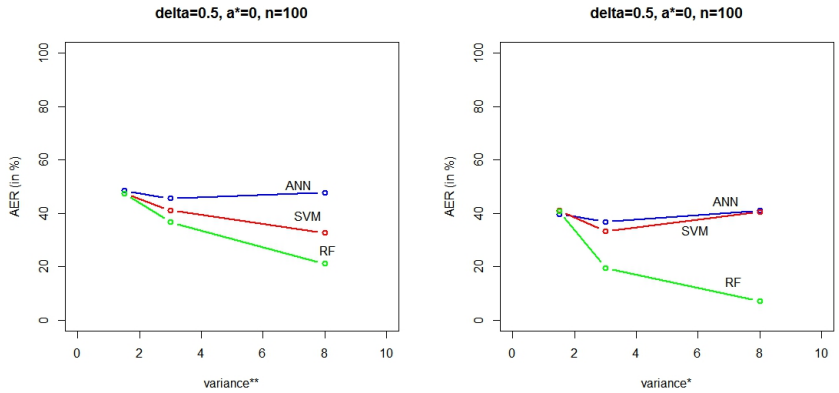


(e) $p = 2$



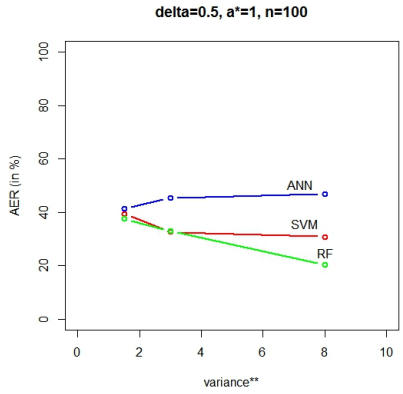
(f) $p = 10$

Figure 2.8: Plots of expected actual error rates of SVM, RF and ANN over simulated index sample for $n = 25$, $p = (2, 10)$ and $\delta = .5$ depicting the effect of variability on error rates.

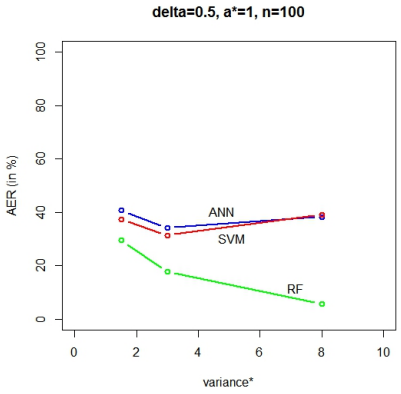


(a) $p = 2$

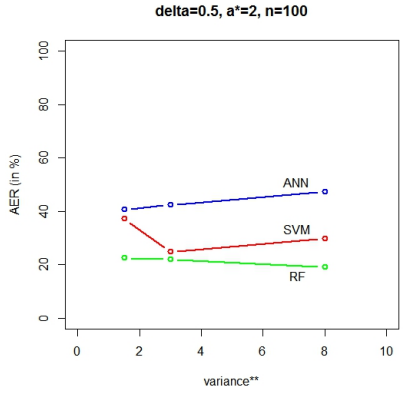
(b) $p = 10$



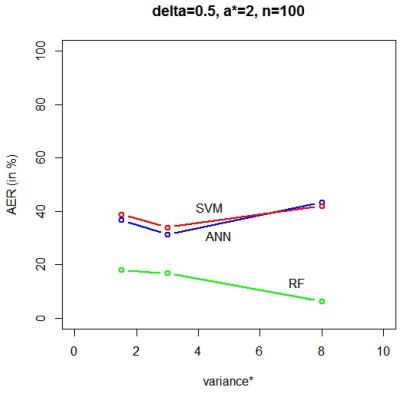
(c) $p = 2$



(d) $p = 10$



(e) $p = 2$



(f) $p = 10$

Figure 2.9: Plots of expected actual error rates of SVM, RF and ANN over simulated index sample for $n = 100$, $p = (2, 10)$ and $\delta = .5$ depicting the effect of data skewness on error rates.

$p=2$				$\delta=.5$											
a	σ^2	S_k		Sample Size											
			$D.F.$	25		50		100		400		600		1000	
				APE	AE	APE	AE	APE	AE	APE	AE	APE	AE	APE	AE
0	1.5	477.43	SVM	40.80	47.24	42.50	47.62	45.53	47.52	45.15	45.95	46.20	45.62	46.07	45.63
			RF	0	49.20	0	51.22	0	47.51	0	47.73	0	47.37	0	47.15
			ANN	44.75	49.16	45.75	48.13	47.54	48.50	47.71	47.60	47.45	47.32	48.05	47.61
1		478.92	SVM	33.73	38.34	36.50	38.17	36.76	39.32	38.20	38	37.84	37.60	37.97	37.65
			RF	0	37.25	0	37.62	0	37.52	0	36.72	0	35.72	0	35.79
			ANN	46.08	44.85	42.42	42.65	40.73	41.45	41.44	41.49	41.85	41.22	42.55	42.16
2		288.42	SVM	33.06	37.05	34.60	36.92	35.25	37.23	35.00	35.96	34.10	34.86	33.63	34.19
			RF	0	22.65	0	21.63	0	22.58	0	21.11	0	21.11	0	21
			ANN	39.17	39.57	40.08	40.84	39.90	40.63	36.89	37.21	39.56	39.78	38.55	38.77
0	3	458.24	SVM	26.53	42.23	33.53	42.04	37.33	41.07	39.75	41.82	40.40	41.85	40.52	42.21
			RF	0	38.21	0	38.05	0	36.60	0	35.70	0	35.35	0	36.08
			ANN	44.83	48.34	45.58	47.37	45.33	45.58	45.41	45.79	46.24	46.55	47.27	47.54
1		607.63	SVM	27.26	34.32	30.46	33.97	25.31	32.62	30.23	31.63	30.93	31.42	31.92	31.87
			RF	0	34.66	0	33.25	0	32.97	0	32.40	0	32.20	0	32.20
			ANN	42.94	44.44	42.29	42.68	44.75	45.27	44.40	44.30	44.84	44.77	43.13	43.14
2		453.63	SVM	13.33	28.23	18.20	26.57	21.50	24.84	25.50	26.30	24.53	24.76	25.92	26.13
			RF	0	23.80	0	23.29	0	22.03	0	21.07	0	20.94	0	20.20
			ANN	40.75	42.79	40.42	41.55	42.77	42.38	43.56	43.67	41.85	41.56	43.20	43.16
0	8	713.55	SVM	19.73	33.86	26.70	32.13	30.70	32.66	36.60	36.60	35.48	35.14	39.54	39.47
			RF	0	24.07	0	21.86	0	21.17	0	20.63	0	20.39	0	20.01
			ANN	47.50	47.13	45.58	46	47.75	47.66	48.73	48.65	48.30	48.64	48.93	49.29
1		516.40	SVM	18.20	31.01	24.03	28.74	28.45	30.75	34.69	34.69	37.25	36.86	38.35	38.21
			RF	0	23.72	0	20.94	0	20.44	0	19.31	0	18.93	0	19.09
			ANN	43.50	43.54	44.92	45.83	46.67	46.83	48.39	48.51	47.71	47.99	48.97	49.20
2		666.76	SVM	18.46	28.17	25.33	28.45	28.41	29.85	32.39	32.70	37.12	37.37	41.47	41.23
			RF	0	21.15	0	19	0	19.12	0	17.68	0	17.88	0	17.56
			ANN	18.46	28.17	25.33	28.45	28.41	29.85	32.39	32.70	37.12	37.37	41.47	41.23

Table 2.5: Misclassification error rates (in %) of SVM, RF and ANN for simulated skewed data for $(p = 2, \delta = 0.5)$

$p=2$				$\delta=.9$											
				Sample Size											
a	σ^2	S_k	$D.F.$	25		50		100		400		600		1000	
				APE	AE	APE	AE	APE	AE	APE	AE	APE	AE	APE	AE
0	1.5	477.43	SVM	32	47.96	35.86	46.71	38.48	46.32	42.22	45.50	42.71	45.46	42.99	45.45
			RF	0	48.46	0	48.19	0	48.3	0	47.42	0	47.36	0	47.30
			ANN	46.58	49.29	46.38	48.38	45.90	48.66	46.33	47.19	46.78	46.71	46.83	46.66
1		478.92	SVM	27.13	33.57	29.26	32.30	30.13	31.61	31.80	30.43	32.11	30.14	32.01	30.09
			RF	0	37.60	0	37.35	0	37.07	0	36.26	0	36.12	0	36.01
			ANN	42.33	44.84	38.42	37.29	38.90	38.43	34.99	33.45	34.42	32.06	34.74	32.90
2		288.42	SVM	20.40	26.28	19.96	24.74	20.73	21.99	19.57	19.02	19.36	18	19.05	17.59
			RF	0	22.79	0	22.11	0	22.63	0	21.40	0	21.42	0	21.20
			ANN	31.75	32.57	27.21	29.05	24.44	24.78	24.25	23.60	23.19	22.38	23.46	22.75
0	3	458.24	SVM	11.53	42.31	16.46	38.58	23.15	36.97	27.95	34.22	29.32	33.18	30.42	32.87
			RF	0	38.07	0	37.69	0	37.05	0	35.65	0	36.07	0	35.97
			ANN	41.08	47.10	42.67	44.43	42.56	44.32	42.63	43.67	42.53	43.24	41.40	42.41
1		607.63	SVM	13.53	34.42	19.30	32.05	22.18	31.12	27.46	29.58	27.08	29.67	28.18	29.05
			RF	0	35.95	0	33.32	0	33.46	0	32.30	0	31.96	0	32.09
			ANN	41.17	42.46	38.21	39.36	37.25	37.77	35.95	35.39	33.54	33.25	33.39	33.30
2		453.63	SVM	10	23.36	15.50	22.79	18.40	20.71	19.59	19.26	19.58	19.24	20.21	19.09
			RF	0	24.88	0	23.11	0	22.62	0	21.75	0	20.88	0	20.69
			ANN 31.75		33.43	33.67	32.22	34.58	34.62	33.44	32.70	28.99	28.10	29.79	28.95
0	8	713.55	SVM	7.06	33.15	12.43	27.94	16.76	24.94	21.69	23.68	23.80	24.09	23.50	23.45
			RF	0	24.43	0	21.90	0	20.98	0	20.59	0	19.97	0	19.69
			ANN	43.58	44.05	38.58	40.22	40.19	40.60	42.33	42.08	41.22	41.50	41.19	41.24
1		516.40	SVM	7.26	30.87	15.10	25.56	20.11	23.52	23.58	24.46	22.66	23.27	24.56	24.70
			RF	0	23.67	0	20.88	0	20.39	0	19.26	0	18.91	0	18.50
			ANN	40	41.45	40.54	41.53	41	40.75	39.71	40	40.40	40.76	42.34	42.51
2		666.76	SVM	9.8	25.85	13.46	21.97	16.15	20.60	19.48	20.25	20.30	20.88	20.22	20.84
			RF	0	20.70	0	19.91	0	18.48	0	18.12	0	17.98	0	17.63
			ANN	44.17	44.20	41.29	41.46	38.31	38.40	43.75	43.85	37.51	37.59	41.57	41.45

Table 2.6: Misclassification error rates (in %) of SVM, RF and ANN for simulated skewed data for ($p = 2, \delta = 0.9$)

$p = 2$			$\delta = .5$					
			Skewed vs Skewed					
a	σ^2	$D.F.$	Sample Size					
			25	50	100	400	600	1000
0	1.5	ANN	.50	.51	.51	.52	.52	.52
		SVM	0.52	0.53	0.54	0.54	0.54	0.54
		RF	.50	.51	.52	.52	.52	.52
1		ANN	.55	.57	.58	.58	.58	.57
		SVM	0.61	0.61	0.60	0.61	0.62	0.62
		RF	.62	.62	.62	.62	.64	.64
2		ANN	.60	.59	.59	.62	.60	.61
		SVM	0.62	0.62	0.64	0.65	0.65	0.65
		RF	.77	.78	.77	.78	.78	.78
0	3	ANN	.51	.52	.54	.54	.53	.52
		SVM	0.57	0.57	0.58	0.58	0.58	0.57
		RF	.61	.61	.63	.64	.64	.63
1		ANN	.55	.57	.54	.55	.55	.56
		SVM	0.65	0.66	0.67	0.68	0.68	0.68
		RF	.65	.66	.67	.67	.67	.67
2		ANN	.57	.58	.57	.56	.58	.56
		SVM	0.71	0.73	0.75	0.73	0.75	0.73
		RF	.76	.76	.77	.78	.79	.79
0	8	ANN	.52	.53	.52	.51	.51	.50
		SVM	0.65	0.67	0.67	0.63	0.64	0.60
		RF	.75	.78	.78	.79	.79	.79
1		ANN	.56	.54	.53	.51	.51	.50
		SVM	0.68	0.71	0.69	0.65	0.63	0.61
		RF	.76	.79	.79	.80	.81	.80
2		ANN	.53	.53	.52	.51	.51	.50
		SVM	0.71	0.71	0.70	0.67	0.62	0.58
		RF	.78	.80	.80	.82	.82	.82

Table 2.7: Average AC1 statistic of ANN, SVM and RF for simulated skewed data with $(p = 2, \delta = .5)$.

3

New Discrminant Function and Methodology for Classification of Highly Skewed Data

3.1 Introduction

The distribution-free approach of the three most popular and the most advanced non-parametric machine learning classification algorithms discussed in Chapter 2 prompted us to critically examine their performance for highly skewed data. The findings suggested that only the ensemble methods based Random forest technique may prove to be robust in accurate and efficient classification of severely skewed datasets. In the present chapter, we take note of the fact that despite of their distribution free approach, these advanced machine learning algorithms lag behind the parametric maximum likelihood classifier (MLC) in terms of wide scale adaptation among practitioners and their frequent application to the classification problems in real life. Hence, in the remaining part of this section i.e. Section 3.1 of this chapter we first highlight the grave concerns that limit the wide scale use of artificial neural networks (ANN), support vector machines (SVMs) and the random forests (RFs) classifiers in the classification field across varied disciplines. And then give an account of the strengths of the MLC which make it the most preferred and the most readily used classification technique among the practitioners and the data analysts. Noticing the lack of any specific study measuring the effect of severe data skewness on the performance of the MLC, we try to fill this gap by attempting the same in the present chapter and lastly propose a new dis-

criminant function and the methodology based on it for efficient classification of severely skewed datasets in Section 3.3. A numerical illustration justifying the proposed discriminant function based methodology is illustrated in Section 3.4 and Section 3.5 discusses the concluding remarks.

3.1.1 Limitations of non-parametric classifiers

In the last chapter, we elaborated the relative advantages of the machine learning algorithms ANN, SVM and RF over the classical MLC with a brief overview of their limitations. In the following paragraphs we discuss in detail some of the major limitations of these classification algorithms that ultimately led us to revive the theoretically robust MLC with some modifications for skewed data.

The ANN is often referred to as a *black box technique* (Qiu and Jensen, 2004) due to the complex nature of the underlying architectures and the outputs of the network which unable its user to accurately understand the processes that translate input features to output classes (Szuster et al., 2011). The most challenging limitation of an ANN is that their performance is highly dependent on how well they are trained by the user in terms of the selection of proper architecture for designing the network and learning its parameters optimally. Moreover, the generalization capability of ANN for unseen datasets is limited by training size considerations, number of nodes and the number of hidden layers used and the amount of time taken for training the network (Atkinson and Tatnall, 1997). For example, designing a small network with an insufficient number of nodes can lead to a poor approximation and generalization by the network. Whereas on the contrary a large network with excessive number of nodes might learn specific properties of the training data thereby making the search for global optimum more difficult and hence, may result in overfitting of the training data (Camargo and Yoneyama, 2001). Apart from these, the relatively larger training times and larger feature sets are other factors that limit the performance of neural networks. Despite certain guidelines suggested in some significant works like, Wilkinson et al. (1995); Garson (1998); Kon and Plaskota (2000) and Kavzoğlu (2001) to optimally set a networks parameters, learning a networks parameters optimally remains a tough task to accomplish for a non-expert user.

Similar to the neural networks, the major setback concerning the efficient

application of SVM classifiers is the complete dependence of their performance on the appropriate learning of the parameters involved which include the kernel functions and their respective parameters. Additionally, from a practical point of view, the high algorithmic complexity and extensive memory requirements of the required quadratic programming in large-scale tasks are the more serious problems of an SVM classifier. And hence, from a non-experts point of view, the underlying theory of SVM may be a bit intimidating. Also, SVM works optimally in a binary class problem and its performance is affected to large in multiclass scenario by unbalanced classes under one-against-all strategy (Pal, 2008) and the huge memory requirements of one-against-one strategies (Hsu and Lin, 2002). Sensitivity to heterogeneous data and to outliers are other factors that deteriorate the performance of SVMs.

Although random forest classifiers do not need much user level expertise in order to be trained, but they do take up a lot of time as multiple trees are grown to generate a forest and usually underestimate the misclassification errors of the training datasets which is also evident from the results obtained in previous chapter. The results generated by RFs are easy to interpret as compared to those of ANN but they too act as *black box techniques* as they do not allow an insight in to examining each individual tree separately.

3.1.2 Motivation, objective and scope of the study

Having discussed limitations of the non-parametric alternatives to MLC and after examining the performances of the most advanced algorithms among the class of non-parametric classifiers for dealing with specific non-normalities in datasets, we reach to the conclusion that these machine learning algorithms can not be the *de facto* choice for handling severely skewed datasets especially for a non-expert user. Also it is evident from vast literatutre of classification methods that no single machine learning algorithm can cater to all types of classification problems and in the absence of any particular guidelines for selecting the best classifier for a specific study, the need is to look out for that algorithm which can fairly work with a larger number of datasets without compromising, significantly, with the classification accuracy (Lu and Weng (2007); Khondoker et al. (2013)). Each and every classification algorithm has its own strengths and limitations. The performance of traditional discriminant analysis techniques under non-optimal conditions had been a topic of debate since long . But in spite of all these years of extensive research and development of

some pretty good and advanced alternative methods of classification, the traditional discriminant analysis techniques continue to be the most frequently used method of classification (Jensen (2005); Thenkabail (2015)) since when these were first used by Fisher to differentiate between the three species of Iris flower (Fisher, 1936). These discriminant functions are widely known as Linear discriminant function (LDF) and Quadratic discriminant function (QDF). The associated assumptions are of common covariance for LDF and of normality of underlying populations with different covariance matrices for QDF. MLC based on linear discriminant function (LDF) and quadratic discriminant function (QDF) have their own benefits in contrast to the machine learning methods. The underlying statistical concepts of these discrminant functions are relatively much simpler and do not need user level expertize for their formulation. The nature of the computations involved is far from complex which provides a clear insight in to the working of these discrminant functions and aids easy interpretation of the classification results. Robust underlying statistical theory, efficient performance for higher dimensional datasets, multi-class as well as very large datasets, faster computations and their inclusion in almost all the image processing softwares are other reasons for which MLC continue to enjoy acceptance amongst the researchers and analysts across varied research fields. Also none of the non-parametric classifiers take into account the parametric information, if available any, about the underlying datasets. This consideration may prove to be highly significant in improving a classifier's performance as when the underlying populations or classes are closed to the assumed parametric distributions, the maximum likleihood classifiers are proven to perform best (Chaudhuri et al. (2009); Fukunaga (2013)) among all the classifiers. Having said that, we discourage the increasing practice of completely overlooking the claiber of these parametric classifiers and hence feel the need to explore the modifications of these discriminant functions for dealing with non-normal data.

The objective of the study in this chapter is to review the previous attempts taken by the researchers to examine the performance of the theoretically robust maximum likelihood classifiers on severely skewed datasets and to suggest a new discriminant function as well as a new methodology which exploits the full calibre of these robust parametric classifiers without compromising with the classification accuracies under non-optimal situations of severe skewness of the datasets. Moreover, as was emphasized in the previous chapter, inspite of a large number of researches produced which compare the performances

of various classifiers with that of MLC on particular but vastly varied real datasets, there are a few exceptions based on simulated data which particularly investigate the performance of MLC on skewed datsets. And since the better performance of any particular classifier on one or a few instances cannot guarantee the same for all the other datasets, hence simulation studies may be a better alternative for objectively and feasibly investigating the performance of classification algorithms (Yousefi et al., 2011a). Hence, we aim at illustrating the performance of the traditional MLC and the proposed discrminant function on skewed datasets using extensively simulated datasets as well as some real datasets. The results obtained in this work are based on simulated datasets, therefore they do not specifically cater to the classification issues of any particular discipline and can be referred to in general for any type of classification problem.

3.2 Background

The use of MLC, even if it is for comparison of classification performance, in almost all the research publications on machine learning algorithms from varied disciplines is itself an evidence of the extent of popularity of it among the researchers. Among a large number of performance measuring studies on MLC there are only a few based on simulated datasets and even fewer on skewed simulated datasets. Instead the performance of the traditional LDF and QDF to non-normal data has been widely examined in the existing literature of comparative studies based on real datasets. Some landmark works that contributed significant investigations measuring the effect of skewness on the LDF and QDF empirically as well as graphically have been discussed in this section. Robustness of LDF and QDF to certain types of non-normality, specifically for lognormal, logit normal and hyperbolic sine normal distributions was investigated in Lachenbruch et al. (1973) and the robustness of the QDF to lognormally distributed data was established using simulation techniques except when the data is highly skewed in Clarke et al. (1979). Misclassification probabilities were plotted as functions of mean and prior probabilities to advocate the use of appropriate transformations to normality while discriminating lognormal data in Beauchamp et al. (1980). An extensive simulation study was conducted on lognormal data to compare the performances of LDF with other methods for classifying lognormal data and logistic function approach was found to be superior to others in Baron (1991). A simulation study was

designed so as to generate realistic gene expression data and also non normal data from Poisson distribution in (Khondoker et al., 2013) and the boundaries, in terms of various factors such as feature to training data ratio, that limit the performance of several parametric and non-parametric classifiers were also reported. A separate discriminant function referred to as skew normal discriminant function (SDF) was suggested in Azzalini and Capitanio (1999) for classifying skewed data but the study was restricted to comparing LDF with SDF for classifying equi-variant bivariate populations. This approach was further extended for multiclass data in Zadkarami and Rowhani (2010). Lachenbruch (1975); Johnson et al. (1979); Dillon (1979); McLachlan (2004) and Seber (2009) are some other initial works that contributed significantly to the study of robustness of MLC to non-normal data and concluded that LDF and QDF are sensitive to deviations from normality.

All these above mentioned works took their attempts on robustness studies of LDF and QDF using simulated skewed data with a specific range of skewness that did not seriously affect the performance of QDF (or, LDF). In the subsequent sections we have investigated the comparative performances of LDF, QDF and the suggested lognormal discriminant function (LNDF) based classifier on severely skewed data.

3.3 Maximum Likelihood Classifier and the Suggested Methodology

In this section, we define briefly the classical maximum likelihood classifier, the proposed lognormal discriminant function (LNDF), a new algorithm based on the LNDF and some other methods needed for the application of the proposed algorithm.

3.3.1 Maximum likelihood classifier

Maximum likelihood classifier is a supervised statistical approach to pattern recognition or classification which is based on the Bayesian classification theory. As the name suggests this classifier allocates an observation to the class with the maximum liklelihood. In other words, MLC calculates the posterior probalibility of an observation belonging to all the predefined set of classes and accordingly allocates it to the one for which it has the highest value of the posterior probability.

If Π_1,Π_2,... Π_m are the m populations or information classes and a $(p \times 1)$ vector $\mathbf{x}$ denotes an observation, then the Bayes classifier or MLC allocates the observation $\mathbf{x}$ to the k^{th} information class if

$$P(\Pi_k|\mathbf{x}) > P(\Pi_i|\mathbf{x}) \qquad \forall \quad i,k = 1,2,\ldots,m, \qquad i \neq k \tag{3.1}$$

where, $P(\Pi_i|\mathbf{x})$ is the class conditional probability that the observation vector $\mathbf{x}$ comes from the class Π_i given that $\mathbf{x}$ was observed and is obtained using the *Bayes probability theorem* as

$$P(\Pi_i|\mathbf{x}) = \frac{P(\mathbf{x}|\Pi_i).p(\Pi_i)}{p(\mathbf{x})} \tag{3.2}$$

where, $p(\Pi_i)$ is the prior probability of the i^{th} information class, $P(\mathbf{x}|\Pi_i)$ is the probability of finding an observation from class Π_i and

$$p(\mathbf{x}) = \sum_{i=1}^{m} P(\mathbf{x}|\Pi_i).p(\Pi_i) \tag{3.3}$$

is the probability of occurence of an observation which remains uniform over all observations and hence, can be ignored for classification so that the classification criterion in equation (3.1) reduces to

$$P(\mathbf{x}|\Pi_k).p(\Pi_k) > P(\mathbf{x}|\Pi_i).p(\Pi_i) \qquad \forall \quad i,k = 1,2,\ldots,m, \qquad i \neq k \tag{3.4}$$

or, equivalently

$$\ln P(\mathbf{x}|\Pi_k).p(\Pi_k) > \ln P(\mathbf{x}|\Pi_i).p(\Pi_i) \qquad \forall \quad i,k = 1,2,\ldots,m, \qquad i \neq k \tag{3.5}$$

where, $\ln P(\mathbf{x}|\Pi_i).p(\Pi_i)$ is generally referred to as *discriminant function.*
The conditional probabilities $P(\mathbf{x}|\Pi_i)$ in equation (3.5) can represent any known statistical probability distribution function. But in the most general setting of a MLC which is incorporated in all image analysis softwares and is used mostly for practical application, these are assumed to be following multivariate gaussian distributions due to obvious reasons of mathematical convenience and reasonably high efficiencies produced by them across a wide variety of population models (Johnson and Wichern, 2007). Hence, with the assumption of gaussian populations these probabilities can be expressed as

$$P(\mathbf{x}|\Pi_i) = \frac{1}{(2\pi)^{p/2}|\ \boldsymbol{\Sigma}_i\ |^{1/2}} \exp\Big[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)'\,{\boldsymbol{\Sigma}_i}^{-1}(\mathbf{x}-\boldsymbol{\mu}_i)\Big] \tag{3.6}$$

with $\boldsymbol{\mu}_i$ and $\boldsymbol{\Sigma}_i$ as the mean and variance-covariance matrix respectively of the i^{th} population. Consequently the maximum liklelihood decision rule in equation (3.5) becomes

$$\mathbf{x} \in \boldsymbol{\Pi}_k, if$$

$$\begin{aligned} \ln P(\mathbf{x}|\Pi_k).p(\Pi_k) &= \ln p(\Pi_k) - \frac{p}{2} - \frac{1}{2}\ln|\ \boldsymbol{\Sigma}_k\ | - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)'\,{\boldsymbol{\Sigma}_k}^{-1}(\mathbf{x}-\boldsymbol{\mu}_k) \\ &= \max_i \ln P(\mathbf{x}|\Pi_i).p(\Pi_i). \end{aligned} \tag{3.7}$$

After ignoring constants, we get

$$d_i(\mathbf{x}) = \ln p(\Pi_i) - \frac{1}{2}\ln|\ \boldsymbol{\Sigma}_i\ | - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)'\,{\boldsymbol{\Sigma}_i}^{-1}(\mathbf{x}-\boldsymbol{\mu}_i) \tag{3.8}$$

where, $d_i(\mathbf{x})$ is the so-called population discriminant function for the i^{th} population. In most practical situations, the population parameters $\boldsymbol{\mu}_i$ and $\boldsymbol{\Sigma}_i$ are not known and hence, are replaced by their maximum likelihood estimates and the above population discrminant score modifies to the sample discriminant score,

$$d_i(\mathbf{x}) = \ln p(\Pi_i) - \frac{1}{2}\ln|\ \boldsymbol{S}_i\ | - \frac{1}{2}(\mathbf{x}-\bar{\mathbf{x}}_\mathbf{i})'\,\boldsymbol{S}_i^{-1}(\mathbf{x}-\bar{\mathbf{x}}_\mathbf{i}). \tag{3.9}$$

Having defined sample discriminant function, now we can formally define two most common discriminant functions, i.e. the QDF and the LDF which are used for allocating an observation into one of the many possible classes in a maximum likelihood classifiers.

3.3.1.1 Quadratic discriminant function

If Π_1,Π_2,... Π_m are the m populations assumed to be coming from p-variate gaussian distribution $N_p(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$, $N_p(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$,...,$N_p(\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)$ respectively, $T_r = \{X_{ij}, i = 1, 2, \ldots m, j = 1, 2, \ldots n_i\}$ is the training dataset where, X_{ij} denotes the p- variate j^{th} observation from the i^{th} population, then the discrminant function in equation (3.9) is referred to as the quadratic discriminant function

and allocates observation $\mathbf{x}$ of p measured features to the population which has the largest sample quadratic discriminant score d_i given as

$$d_i(\mathbf{x}) = -\frac{1}{2}\ln \mid \boldsymbol{S}_i \mid -\frac{1}{2}(\mathbf{x}-\bar{\mathbf{x}}_\mathbf{i})' \boldsymbol{S}_i^{-1}(\mathbf{x}-\bar{\mathbf{x}}_\mathbf{i}) + \ln p_i. \tag{3.10}$$

Where, p_i is the prior probability that an observation belongs to the i^{th} population, $\bar{\mathbf{x}}_i = \frac{1}{n_i}\sum_{j=1}^{n_i} X_{ij}$ and $\boldsymbol{S_i} = \frac{1}{n_i-1}\sum_{j=1}^{n_i}(X_{ij}-\bar{\mathbf{x}}_i)(X_{ij}-\bar{\mathbf{x}}_i)'$ are the maximum likelihood estimates of mean vector ($\boldsymbol{\mu}_i$) and the variance-covariance matrix ($\boldsymbol{\Sigma}_i$) respectively of the i^{th} population estimated from a random training sample of size n_i.

3.3.1.2 Linear discriminant function (LDF)

If all the m gaussian populations Π_1,Π_2,... Π_m are homogeneous with respect to their variance-covariance matrices, such that $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2 = \ldots = \boldsymbol{\Sigma}_m = \boldsymbol{\Sigma}$ then the discriminant function in equation (3.9) is referred to as the linear discriminant function (LDF) and assigns the observation $\mathbf{x}$ to the population which maximizes the linear discriminant score d_i given as

$$d_i(\mathbf{x}) = -\frac{1}{2}\ln \mid \boldsymbol{S} \mid -\frac{1}{2}(\mathbf{x}-\bar{\mathbf{x}}_\mathbf{i})' \boldsymbol{S}^{-1}(\mathbf{x}-\bar{\mathbf{x}}_\mathbf{i}) + \ln p_i \tag{3.11}$$

where, $\boldsymbol{S} = \frac{1}{\sum_{i=1}^{k} n_i - k}\left(\sum_{i=1}^{k}(n_i-1)\boldsymbol{S_i}\right)$ is the pooled unbiased estimate of population variance covariance matrix $\boldsymbol{\Sigma}$.

The linear discriminant function in equation (3.11) turns out to be of the same form as that of Fisher's linear discriminant function. Fisher used a distribution-free approach based on taking linear combinations of $\mathbf{x}$ to reach at equation (3.11) and hence, LDF can be used without assuming gaussian distribution for the underlying populations. However, LDF is found to be asymptotically optimal in terms of producing minimum misclassification probabilities when the underlying populations are normal with equal covariance matrices (Rencher, 2003).

It should be noted here that for using QDFs based on $\mathbf{S}_i$, the sample sizes n_i must be greater than the number of features (p) as well as should be large enough in order to obtain stable estimates of the population parameters. LDF however does not impose any such restrictions on the sample sizes as it requires the estimation of lesser number of parameters. The prior probabilities, p_i of the populations if unknown, are assumed to be uniform over all populations generally. However, a suitable modelling of $p_i's$ as weighted priors (Strahler

(1980); Mather (2004)) or as smoothness priors (Magnussen et al. (2004); Tso and Olsen (2005)) can aid in improving the performance of a MLC while classifying digital images.

3.3.2 Suggested methodology

A software based automatic classification procedure uses discriminant functions while testing the underlying populations for normality and using suitable transformations, if populations are found to be non-normal, are left to be employed manually by the user or analyst. The choice of a normality test and of a normality transform requires some degree of statistical knowledge which, the analyst, may not be equipped with, for instance, a biologist analyzing a microarray data, a meteorologist analyzing a digital image etc. Therefore, a mechanism addressing skewness in data is needed to be incorporated in the discriminant analysis procedure in a software package, thereby avoiding manual intervention for choosing an appropriate transformation. Hence, a discriminant function based on multivariate lognormal distribution is proposed here, to analyze skewed data. It can easily be implemented in a machine based automatic classification mechanism.

3.3.2.1 Lognormal discriminant function (LNDF)

The lognormal distribution is flexible enough to take on quite different shapes for different values of the parameters in the simple basic distribution formula which makes it capable of describing many forms of experimental data which are asymmetrically distributed. The use of the distribution in such cases of skewness, particularly of severe skewness will give more meaningful results than the common assumption of a normal distribution, and it is recommended that asymmetrically distributed data should always first be tested to see whether they conform to the lognormal distribution (Aitchison and Brown (1963); Gale (1967); Crow and Shimizu (1988)). Thus, when the data is found to be skewed in nature then instead of assuming the underlying populations as multivariate normal, the proposed discriminant function LNDF assumes them to be following multivariate lognormal distribution i.e. $\Pi_i \sim LN_p(\boldsymbol{\mu_i}, \boldsymbol{\Sigma_i})$. Now using Bayes rule, LNDF allocates the observation $\mathbf{x}$ to the population which has the largest posterior probability $P(\Pi_i \mid \mathbf{x})$, i.e. $\mathbf{x}$ is allocated to the k^{th}

population if

$$\ln p_k g_k(\mathbf{x}) \geq \ln p_i g_i(\mathbf{x}) \qquad \forall \quad i,k = 1,2,\ldots m, i \neq k. \tag{3.12}$$

Where, $g_i(\mathbf{x})$ is the density function of the i^{th} population, the form of which is given as under

$$g_i(\mathbf{x}) = \frac{1}{(2\pi)^{p/2} \mid \boldsymbol{\Sigma_i} \mid^{1/2} x_1.x_2 \ldots x_p} \exp\{-(\ln \mathbf{x} - \boldsymbol{\mu_i})' \boldsymbol{\Sigma}^{-1} (\ln \mathbf{x} - \boldsymbol{\mu_i})/2\}. \tag{3.13}$$

When $\boldsymbol{\mu_i}$ and $\boldsymbol{\Sigma_i}$ are unknown, the density function is estimated as

$$\hat{g}_i(\mathbf{x}) = \frac{1}{(2\pi)^{p/2} \mid \boldsymbol{S_i} \mid^{1/2} x_1.x_2.\ldots x_p} \exp\{-(\ln \mathbf{x} - \bar{\mathbf{x}}_\mathbf{i})' \boldsymbol{S_i}^{-1} (\ln \mathbf{x} - \bar{\mathbf{x}}_\mathbf{i})/2\} \tag{3.14}$$

where, $\bar{\mathbf{x}}_\mathbf{i} = \frac{1}{n_i} \sum_{j=1}^{n_i} \ln X_{ij}$ and $\boldsymbol{S_i} = \frac{1}{n_i - 1} \sum_{j=1}^{n_i} (\ln X_{ij} - \bar{\mathbf{x}}_\mathbf{i})(\ln X_{ij} - \bar{\mathbf{x}}_\mathbf{i})'$ are the maximum likelihood estimates of $\boldsymbol{\mu_i}$ and $\boldsymbol{\Sigma_i}$ respectively. Replacing $\hat{g}_i(\mathbf{x})$ from equation(3.14) in inequality(3.12), the decision rule reduces to maximizing the discriminant score,

$$d_i(\mathbf{x}) = -\frac{1}{2} \ln \mid \boldsymbol{S_i} \mid - \frac{1}{2} (\ln \mathbf{x} - \bar{\mathbf{x}}_\mathbf{i})' \boldsymbol{S_i}^{-1} (\ln \mathbf{x} - \bar{\mathbf{x}}_\mathbf{i}) + \ln p_i. \tag{3.15}$$

When $\Pi_i's$ are homogeneous with respect to their variance-covariance matrices Σ_i, the discriminant score modifies to

$$d_i(\mathbf{x}) = -\frac{1}{2} \ln \mid \boldsymbol{S} \mid - \frac{1}{2} (\ln \mathbf{x} - \bar{\mathbf{x}}_\mathbf{i})' \boldsymbol{S}^{-1} (\ln \mathbf{x} - \bar{\mathbf{x}}_\mathbf{i}) + \ln p_i \tag{3.16}$$

where, $\boldsymbol{S} = \frac{1}{\sum_{i=1}^{m} (n_i - k)} \left(\sum_i^m (n_i - 1) \boldsymbol{S_i}\right)$ is the pooled sample variance covariance estimate.

Testing of normality of multivariate data is a crucial task in classification problems if one prefers to use MLC for classification and many authors have emphasized on the need to perform tests of multinormality before applying parametric maximum likelihood classification procedure(McLachlan, 2004). However, in practice that rarely happens, often due to the lack of user's expertize on the subject and the user often ends up employing the default methods for the classification task. It should be noted here, that MLC in most of the software packages (statistical or non-statistical), including MATLAB uses the LDF as the default discriminating criterion for discrimination between several populations for all types of data, i.e. normal or non-normal. Keeping in mind the

need to treat skewed data differently and to make normality testing an integral part of the MLC mechanism, we have suggested an algorithm here, elaborated in Figure 2.1, to treat positively skewed data more efficiently. This algorithm instead of using LDF (or, QDF) as black box techniques for any data, first tests each class of the data in question for skewness using Mardia's test (Mardia, 1970, 1974, 1980) and then accordingly decides to use LDF (or, QDF) or the suggested LNDF and hence, results in better and more accurate classification outcomes. The discussed algorithm was generated with MATLAB codes.

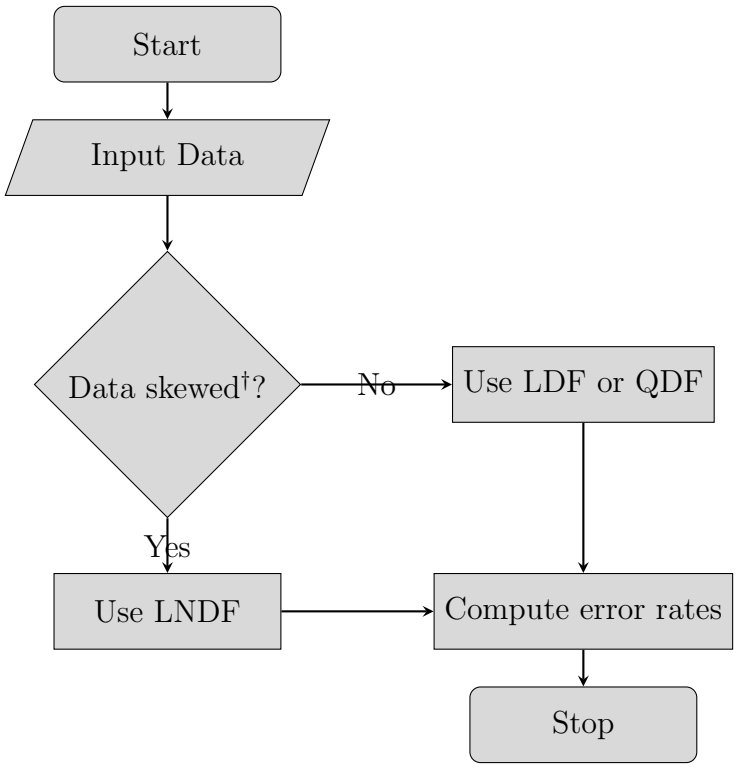


Figure 3.1: Suggested automated classification mechanism.

3.3.3 Mardia's test

The software based automated classification mechanism suggested in Section 3.3.2 requires the data to be tested for multivariate normality in terms of skewness. Moreover, testing for departures from multinormality aids in making practically wiser choices between competing methods of classification. There are many tests in the multivariate literature devoted to this task. A fair review of some of the most significant tests can be found in Srivastava (2002); Mecklin

†Positively Skewed

and Mundfrom (2004) and Hanusz and Tarasińska (2014). Unfortunately there is no single uniformly most powerful test of multinormality in the literature. However, the best known and the most widely used of them is the Mardia's test based on the measures of multivariate skewness and kurtosis which allows to test the hypothesis that conforms with the assumption of multinormality. The multivariate measure of skewness and kurtosis have been defined as natural extensions of univariate measures suggested in Mardia (1970). As we particularly target the non-normality due to skewness of datasets in the present study, Mardia's test based on measures of skewness only have been employed here.
If $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n$ is a random sample of size n from a p-variate distribution with sample mean vector $\bar{\mathbf{x}}$ and sample variance-covariance matrix $\boldsymbol{S}$, then Mardia's p-variate skewness coefficient is defined as

$$b_{1p} = \sum_i \sum_j \left[\left(\mathbf{X}_i - \bar{\mathbf{X}} \right)' \boldsymbol{S}^{-1} \left(\mathbf{X}_j - \bar{\mathbf{X}} \right) \right]^3. \tag{3.17}$$

where, Mardia's multivariate coefficient of skewness, b_{1p} is clearly expressed as the function of standardized third moment.
The Mardia test based on b_{1p} rejects the hypothesis of insignificant skewness for large values of the test statistic,

$$A = n * b_{1p}/6, \tag{3.18}$$

where, A has asymptotic chi-square distribution with $p\,(p+1)\,(p+2)\,/6$ degrees of freedom under normality of $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n$.

3.3.4 Accuracy assessment

It is very obvious that any classification method results in some misclassifications and a measure of the probabilities of these misclassification in the form of error rates of classifiers play a vital role in assessing the performance of a discriminant function in future samples (Johnson and Wichern, 2007). An optimal classifier is the one which results in fewer misclassifications or the one which minimizes the total probability of misclassification (TPM) associated with a discriminant function, given as

$$TPM = \sum_{i=1}^{m} p_i \int_{R_j} f_i(\mathbf{x}) dx \qquad \forall \quad j = 1, 2, \ldots m, \qquad i \neq j \tag{3.19}$$

where, p_i is the prior probability of the ith class, $f_i(\mathbf{x})$ is the probability density funnction of the ith class and R_j is the classification region associated with the jth population whose optimal value is determined as

$$R_j = \frac{f_j(\mathbf{x})}{f_i(\mathbf{x})} \geq \frac{p_i}{p_j} \tag{3.20}$$

in order to minimize TPM. The minimum value of the TPM calculated for optimal $R_j's$ is called as the optimum error rate (OER) of the discrminant function and provides a measure of the global performance of a discriminant function. The OER of a discriminant function is estimated using known population parameters which is usually not the case in practical situations. In case of unknown population parameters, the OER are estimated in terms of the error rates of the sample classification functions which are defined as,

$$\hat{OER} = \sum_{i=1}^{m} p_i \int_{\hat{R}_j} f_i(\mathbf{x})dx \qquad \forall i = 1, 2, \ldots m, \qquad i \neq j \tag{3.21}$$

where, $\hat{R}_j$ is the classification region associated with the jth class and is determined by samples of size n from the jth population. In practice, an unbiased estimate of these optimum error rates associated with a discriminant function are calculated using two quantities which are closely related to the $\hat{OER}$ and can be easily obtained from a *confusion matrix*. These quantities are called as expected apparent error rates (APER) and expected actual error rates (AER) which are obtained by averaging the proportions of observations of training data and testing data respectively that are misclassified by the repeatedly trained sample discriminant functions . To assess the overall performances of the various discriminant functions uesd in this study, the Apparent error rate (APER) as well as the Actual Error rate (AER) which tends to be better estimate of misclassification probabilities were obtained for all the simulated datasets. Apart from overall misclassification rates, measures of chance agreement were also calculated using Gwet's AC1 statistic (Gwet, 2014) in order to assess the reliability of the classifers.

3.4 Numerical Illustration and Results

3.4.1 Simulation and data generation

This section illustrates the optimality of the suggested methodology and the suggested discriminant function LNDF in dealing with severely skewed datasets more efficiently than the regular MLC used in most of the image analysis softwares. For general comparison of the discriminant functions under a variety of distributional settings, an extensive simulation has been conducted here. The following two different situations of class distributions were considered for asseessing the performance of the traditional and the suggested discrminant functions:

- Skewed vs Skewed
- Normal vs Skewed.

Under the first setting, the skewed datasets were simulated in the same manner as described in the previous chapter where, we start off by simulating training datasets from two populations, transforming them to skewed ones and then employ them for training maximum likelihood classifiers based on the LDF, QDF and on the suggested methodology. The first population is kept fixed and the second one is generated with the varied combinations of parameter values given in Table 2.1. In a similar fashion testing samples of size 800 were generated from each of the two populations and the trained classifiers were used to classify these testing samples. Finally, the rates of misclassification of training datasets as well as of testing datasets and the AC1 statistic for the testing dataset were calculated for each of the three classifiers. This whole process of training and testing the classifiers was repeated 30 times and the measured errors and AC1 statistic were averaged over all the replications to get unbiased estimates of the AC1 statistic and of misclassification errors in the form of apparent error rates (APER) and actual error rates (AER) which are tabulated in Tables 3.2 and 3.3. The objective of keeping the simulation design similar to that of the previous chapter is to make the two studies comparable.

Under the second situation, the training and test datasets are simulated using the similar values of the population parameters and the other parameter combinations as were used under the first situation with the only difference as that the distribution of the first class in the training as well as the test dataset

was kept as normal and only the observations of the second class were transformed to the skewed ones. The error rates obtained from the three classifiers under this setting over 30 replications are tabulated in Table 3.4 for bivariate datasets.

3.4.2 Real datasets used

To illustrate the performances of the three discrminant functions discussed in this chapter on the real datasets as well as to compare them with the performances of the machine learning algorithms which were studied in Chapter 2, the real datasets (*Dataset1, Dataset2, Dataset3, Dataset4*) used in this study were kept same as those were used in the previous chapter, described in Section 2.4.2. These are the Landsat dataset, the Indian Liver Patient Dataset (ILPD) and the new thyroid dataset from UCI repository as well as the SPOT dataset (Glasbey, 1988). Similar to the previous chapter, the validation errror rates for the ILPD and the new thyroid dataset were calculated using the leave-one-out method of Lachenbruch (1975) while for the other two datasets, the classifiers were trained and validated using separate training and testing datasets simultaneoulsly. Please refer to Section 2.4.2 for recalling the strategy used for construction of the training and testing datasets and skewness parameters of different classes of the four datasets.

3.4.3 Results

3.4.3.1 Results on simulated data

Based on the results of simulation study conducted in the present chapter which are tabulated in Tables 3.2, 3.3, 3.4 and 3.5, a graphical representation of comparative performances of the three classifiers discussed in the present chapter, along with those of ANN, SVM and RF with respect to the actual error rates against other data characteristics is depicted in Figures 3.2, 3.3 and 3.4. The plots are not shown for all the simulated datasets but we have chosen to plot the error rates only for $n = (25, 100 \text{ and } 600)$ when $p = (2, 10)$ as these vlaues of sample sizes show some changes in the trends of error rates from different classifiers. Figure 3.4 depicts the effect of separability of data classes on the performance of the three discriminant functions. We can infer the following from these plots and tables.

- A look on the misclassification rates of LDF, QDF and LNDF based

methodology given in Tables 3.2 and 3.3 indicates that in all settings, the simulated datasets LNDF based proposed classifer was optimal as it resulted in quite lower misclassification rates as compared to the other two discriminant functions.

- *Effect of training data size:* Sample size considerations were found to be important as it is evident from Tables 3.2, 3.3 and 3.4 and Figure 3.4 that the misclassification error rates show an inverse proportionality to the sample size for all the three discrminant functions under study. Across all the variations in sample sizes considered here, LNDF emerged as a clear winner in terms of producing smaller misclassification error rates except for small sample size under high dimensional setting ie. for $(p = 10, n = 25)$.

- *Effect of Skewness:* For lower ranges of skewness in datasets i.e. for datasets generated with $\delta = (2 \text{ and } 5)$, (results not shown here), QDF was found to be fairly robust as was established in Clarke et al. (1979) and LNDF performed comparably equally well. But as the value of δ drops below 1, i.e. for $\delta = (.5 \text{ and } .9)$ the skewness of the marginal distribution of second population increases, and hence, the gap between the misclassification probabilities obtained from the three discriminant functions starts increasing. It can be seen from Tables 3.2, 3.3 and 3.4 that in all the situations of severe positive skewness in the data, performance of the LNDF was fairly better than that of the QDF (or, LDF). Moreover, as the levels of skewness were increased by increasing the values on the diagonal of covariance matrix of second population, LNDF outplayed the other three classifiers with a clean majority. Apart from this, all the classifiers showed the obvious trends of improved performance with the increase in the separability between the classes which is adjusted with the values of $a^{\dagger}$ in the simulations, as shown in Figure 3.4.

- *Effect of dimensionality:* It can be observed from Figure 3.3 and 3.4 that the overall performances of LDF, QDF and LNDF were found to be improving as the dimensionality of the datasets was increased owing to the fact that added dimensions provide extra information about the datasets in the form of input to the classifiers and hence, aid in better learning

$^{\dagger}a$ is the mean of the first variable of second population,which ensures the variation in the separation between the two populations

of the classifiers. For smaller dimensions of the data, i.e. for $p = 2$, the performance of the suggested MLC was found to be the best in terms of producing lower misclassification rates over all the simulated datasets. But as the dimensions of the feature set increase from $p = 2$ to $p = 10$, suggested classifier was outperformed by the MLC classifiers based on LDF and QDF respectively, for small sample sizes (25 here). For all the other sample sizes considered in the study, LNDF performed exceptionally good for higher dimensional datasets with error rates as low as 1.60%

- *Chance agreement measures:* The values of Gwet's AC1 statistic for LNDF was observed to be lying in the range $(0.5, 0.8)$ and $(.8, .9)$ with higher values for the highly skewed datasets under the *Skewed vs Skewed* and the *Normal vs Skewed* settings respectively, which reports an overall fair to excellent level of agreement. QDF also reported comparable values of the AC1 statistic as shown in Table 3.5. However, LDF reported a constant value of .5 for the chance agreement measure. The comparatively poor performance of LDF over all the datasets may be attributed to the heterogeneous class distributions setting in the present simulation study which points towards the high levels of risk of using LDF without checking the heterogeneity of the populations, beforehand.

- For *symmetric vs skewed* population cases, all the three discriminant functions showed improved performances in terms of error rates as well as of chance agreement measures as compared to the *skewed vs skewed* population cases. This implies that the performance of the MLC is inversely proportional to the number of skewed populations in a dataset (see Table 3.4).

- *Comparison with ANN, SVM and RF:* A look at Figures 3.2, 3.3 and the Tables containing error rates of the previous chapter, i.e. Tables 2.5 and 2.6, reveals that LNDF also outperformed the advanced machine learning classifiers too with a clean majority over all datasets except for moderately skewed datasets when $\Sigma = 1.5(I)$ where, the performance of LNDF lagged behind that of its non-parametric counterparts.

3.4.3.2 Results on real datasets

The following findings were observed on the four real datasets, for their details please see Section 2.4.2.

- For the *Dataset 1*, Landsat dataset, QDF resulted in 15.50% error rate, LDF resulted in 17.85% of error rate and LNDF based classiifer generated 15.55% of error rate. Which implies that LNDF and QDF performed comparably equal and better than the LDF. However, when compared with the performances of ANN, SVM and RF, ANN performed best among all the parametric and non-parametric classifiers with of error rate.

- For the *Dataset 2*, SPOT dataset which has highly imbalanced classes, LDF performed poorly with 78.77% of error rates and LNDF emerged as the best performer with 40.46% of error rates. Comparing the results of this chapter with those of the last chapter on this dataset, only SVM performed marginally better than the proposed classifier.

- For *Dataset 3* too, among the three classifiers, the proposed classifier LNDF performed better than LDF and QDF. But when compared to the performance of machine learning algorithms, ANN emerged as the best performer with only .47% of error rates for this dataset.

- For *Dataset 4* i.e. the ILPD, again LNDF resulted in lower error rates as compared to LDF and QDF but was outperformed by all the three machine learning algorithms.

	LDF		*QDF*		*LNDF*	
	APER	AER	APER	AER	APER	AER
Dataset 1	9.65	17.85	11.50	15.50	6.05	15.55
Dataset 2	75.13	78.77	9.75	45.11	15.98	40.46
Dataset 3	5.58	6.05	5.58	4.19	6.05	2.79
Dataset 4	35.23	36.27	35.92	45.08	32.12	32.82

Table 3.1: Apparaent error rate, Actual error rate of LDF, QDF and proposed LNDF based classifier for the real datasets.

The overall findings observed on the real datasets indicate the better performance of the suggested classifier in 3 out of 4 real datasets as compared to the LDF and QDF. As was emphasized in the previous chapter this is due to

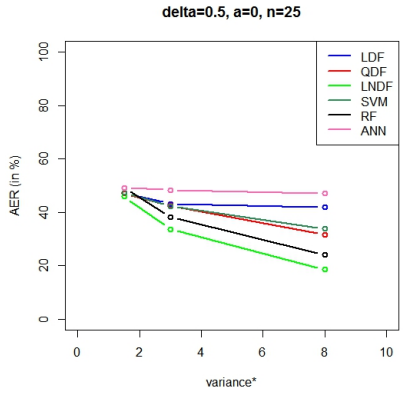
the fact that the real datasets considered were not found to be highly skewed in nature and hence, the results obtained on real datasets do not confirm to those of simulation study.

3.5 Conclusion

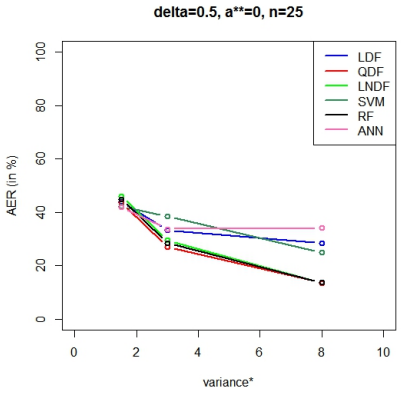
Results of the investigations on simulated data provide empirical evidences that the suggested parametric classification algorithm based on LNDF, the discriminant function obtained using multivariate lognormal distribution, substantially reduces the misclassification of severely positively skewed data as compared to LDF and QDF, and hence should be preferred over the regular one based only on QDF or LDF while dealing with positively skewed data. Furthermore, the performance of the suggested parametric classification algorithm outplayed that of the advanced non-parametric classifiers based on SVM and RF for the simulated data. Therefore, we conclude that in machine based automatic classification mechanism, the availability of LNDF along with the previously available LDF and QDF can avoid the human intervention, such as choosing an appropriate transform and hence fastens the analysis. However, the results of analysis on real datasets suggest that when the choice is to be made between LNDF, SVM and RF, there is no single "best for all" option. But, the added benefits of being computationally much faster than the RFs and lesser complex underlying theoretical concepts, in contrast to the non-parametric advanced classifiers like SVM and RF, which are far more complex to be understandable by a non-expert user makes LNDF a more feasible and viable choice than SVM or RF for very large datasets and for non-expert users. Furthermore, supporting the approach of Chaudhuri et al. (2009) and on the basis of the results obtained above, we argue that the inclusion of the information of the parametric structure of the datasets can significantly improve a classifier's performance and hence, traditional parametric classifiers like MLC should not be overlooked. And further researches suggesting more robust transformations capable of transforming non-normal data to approximate normality should be conducted in order to improve the classical lesser complex and computationally efficient parametric classifiers. We also feel that the incorporation of such improved parametric classifiers, which can automatically handle the non normal data, in the classification toolboxes of some frequently used softwares is needed as it will definitely benefit the non-expert user community.

The methodology suggested in this chapter was shown to be effective enough

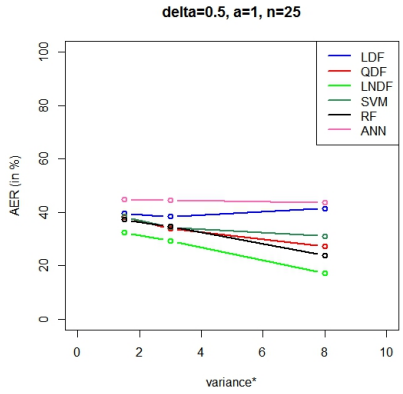
in classifying skewed data with high levels of positive skewness. But further research is required to make efficient use of the capabilities of the MLC for tackling negatively skewed data as well.



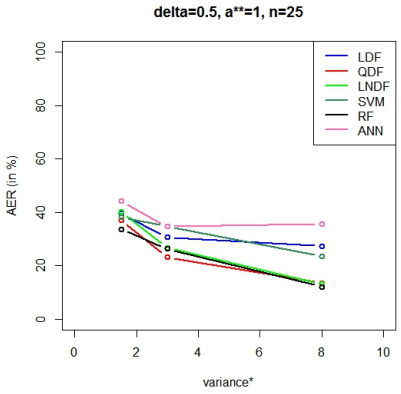
(a) $p = 2$



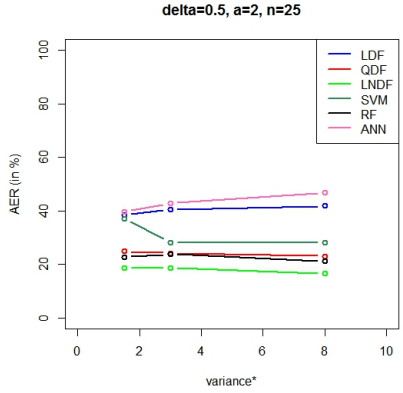
(b) $p = 10$



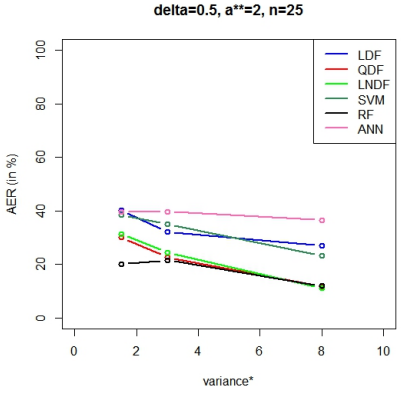
(c) $p = 2$



(d) $p = 2$

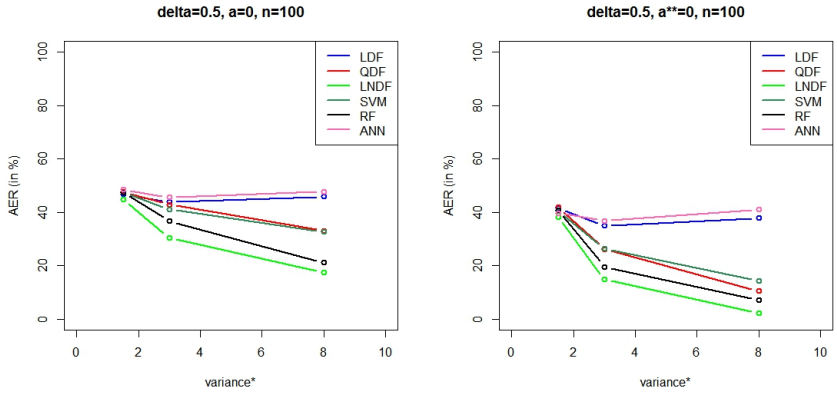


(e) $p = 10$

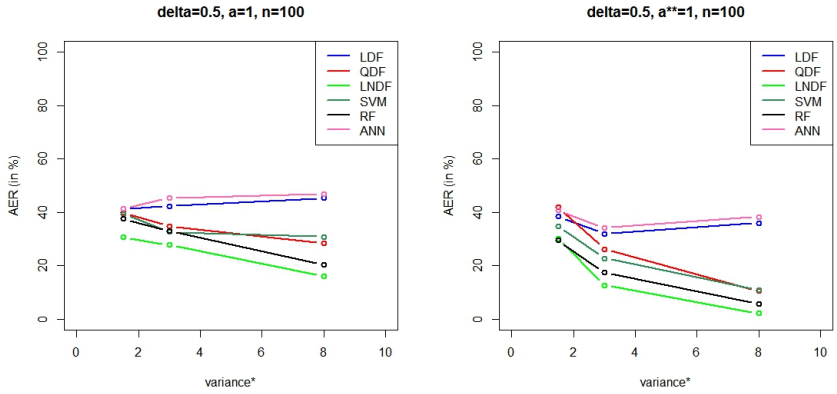


(f) $p = 2$

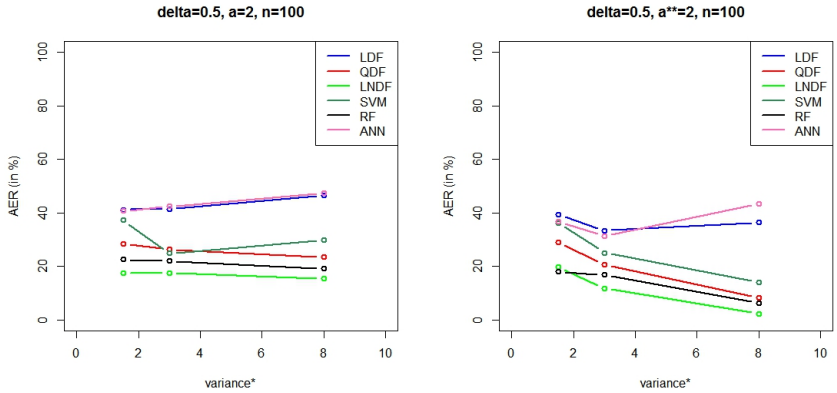
Figure 3.2: Plots of expected actual error rates of LDF,QDF,LNDF,SVM,RF and ANN over simulated index sample for $n = 25$, $p = (2, 10)$ and $\delta = .5$ depicting the effect of data skewness on error rates.



(a) $p = 2$ (b) $p = 10$

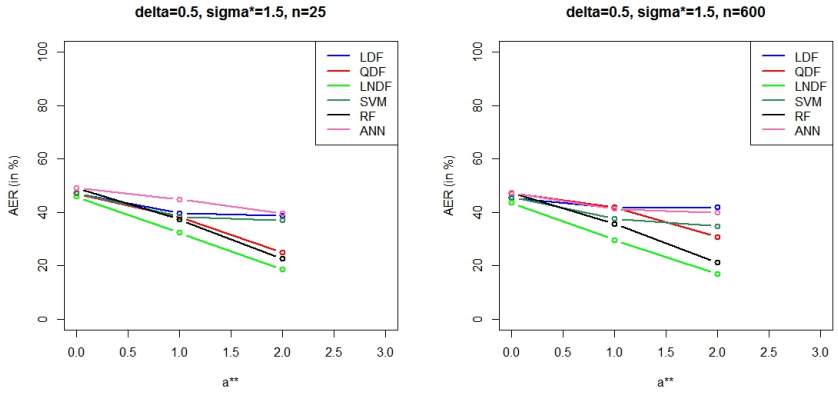


(c) $p = 2$ (d) $p = 2$



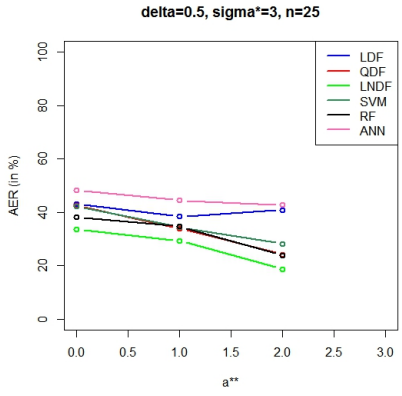
(e) $p = 10$ (f) $p = 2$

Figure 3.3: Plots of expected actual error rates of LDF,QDF,LNDF,SVM,RF and ANN over simulated index sample for $n = 100$, $p = (2, 10)$ and $\delta = .5$ depicting the effect of data skewness on error rates.

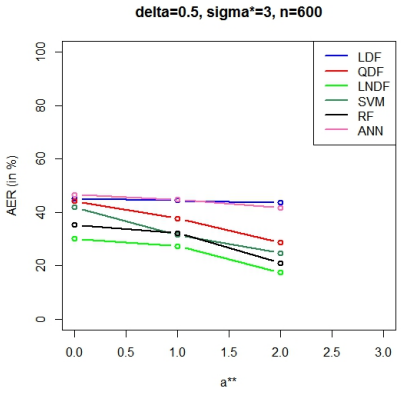


(a) $p = 2$

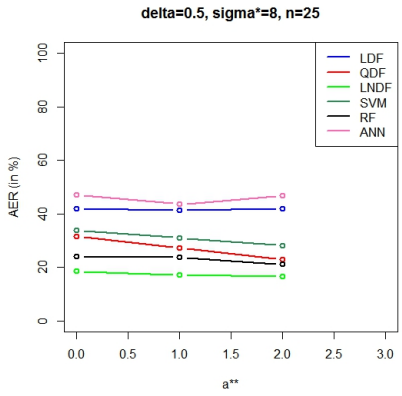
(b) $p = 2$



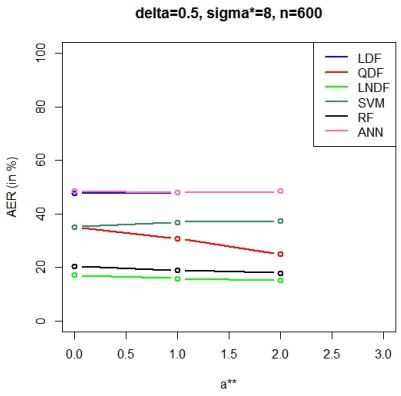
(c) $p = 2$



(d) $p = 2$



(e) $p = 2$



(f) $p = 2$

Figure 3.4: Plots of expected actual error rates of SVM, RF and ANN over simulated index sample for $\delta = .5$ depicting the effect of separability on error rates .

$p=2$			$\delta=.5$											
a	σ^2	$D.F.$	Sample Size											
			25		50		100		400		600		1000	
			APE	AE	APE	AE	APE	AE	APE	AE	APE	AE	APE	AE
0	1.5	LDF	43.93	47.05	45.03	46.79	46.78	46.71	46.53	45.55	46.51	45.25	46.44	45.38
		QDF	43.80	46.98	46.23	46.67	48.08	47.67	47.68	47.05	47.63	47.11	48.03	47.47
		LNDF	37.86	45.84	38.50	45.31	41.65	44.83	42.18	43.06	42.60	43.54	42.34	43.54
1		LDF	38.20	39.56	40.40	40.04	41.50	41.22	41.65	40.99	41.62	41.54	42.26	41.70
		QDF	37.53	38.26	38	37.91	40.28	39.90	42.83	41.77	42.26	41.96	43.18	42.61
		LNDF	28.73	32.29	29.86	31.36	31.11	30.59	31.56	29.82	31.46	29.68	31.36	29.44
2		LDF	38.06	38.83	38.83	39.85	40.83	41.17	41.54	42.11	41.12	41.78	41.40	41.96
		QDF	23.73	24.82	26.73	27.75	27.38	28.46	31	31.19	30.13	30.67	32.10	32.28
		LNDF	16.46	18.52	18	17.96	16.86	17.34	17.90	16.75	18.02	16.75	18.09	16.63
0	3	LDF	41.06	43.20	42.06	43.22	43.06	43.86	43.99	45.10	44.30	45.07	44.73	45.12
		QDF	39.86	42.40	40.63	42.39	42.13	42.91	42.62	44.00	43.23	44.17	43.45	44.27
		LNDF	29.40	33.57	30.20	30.97	30.68	30.33	30.20	29.92	30.48	30.01	30.63	30.15
1		LDF	39.20	38.33	40.86	41.66	42.41	42.27	43.67	44.31	43.87	44.41	44.45	44.91
		QDF	35.06	33.87	34.43	34.32	35.15	34.77	37.03	36.95	37.69	37.66	38.21	38.05
		LNDF	25.13	29.34	26.83	28.20	26.11	27.71	26.22	27.32	27.08	27.29	27.10	27.29
2		LDF	38.60	40.37	40.83	40.89	41.93	41.42	44.10	43.66	43.81	43.64	44.44	44.37
		QDF	23.33	24.15	24.66	24.67	26.76	26.29	28.92	28.71	28.94	28.72	29.59	29.54
		LNDF	19.53	18.73	18.10	17.87	19.55	17.53	19.20	17.30	19.20	17.35	18.97	17.37
0	8	LDF	40.60	41.81	43.96	44.32	45.08	45.88	47.58	47.96	47.44	47.64	47.98	48.16
		QDF	30.13	31.69	33.23	32.32	33.33	33.04	33.92	33.99	35.15	34.96	34.60	35.17
		LNDF	16.73	18.51	17.96	17.60	17.10	17.45	17.53	17.07	17.48	17.07	17.28	16.98
1		LDF	40.26	41.45	42.70	43.27	44.43	45.36	47.32	47.74	47.56	48.04	48	48.27
		QDF	25.06	27.12	28.56	28.22	27.98	28.31	30.07	29.95	30.95	30.83	30.96	30.86
		LNDF	15.93	17.22	16.56	16.17	16.11	15.95	16.80	15.84	16.59	15.84	16.86	15.85
2		LDF	41.53	41.82	44	44.72	45.16	46.39	47.09	47.84	47.68	48.40	47.96	48.71
		QDF	20.06	23.04	22.90	23.35	23.18	23.45	24.93	24.72	25.05	24.95	25.50	25.43
		LNDF	13.40	16.61	14.16	15.76	14.36	15.49	14.29	15.20	14.58	15.12	14.62	15.08

Table 3.2: Expected Apparent and Actual error rates (in %) of LDF, QDF and LNDF for simulated skewed data for $p=2, \delta=.5$ under Skewed vs Skewed population setting.

$p = 10$			$\delta = .5$											
			Sample Size											
a	σ^2	$D.F.$	25		50		100		400		600		1000	
			APE	AE	APE	AE	APE	AE	APE	AE	APE	AE	APE	AE
0	1.5	LDF	30.93	43.82	36.33	42.58	39.22	41.58	41.04	40.56	40.77	40.83	41.31	41
		QDF	25	43.60	36.03	42.60	39.72	41.85	43.32	42.47	43.58	42.88	44.42	43.22
		LNDF	9.27	45.99	17.70	41.90	23.10	38.27	30.22	35.51	30.86	34.54	31.88	34
1		LDF	29.27	39.50	33.57	38.70	36.63	38.40	38.63	38.13	38.98	38.79	39.50	38.15
		QDF	21	36.94	31.33	35.12	33.78	35.40	39.40	38.37	40.13	39.51	40.91	39.77
		LNDF	8.67	40.25	15.63	34.41	18.32	30.26	25.07	25.93	24.86	25.47	25.75	25.15
2		LDF	27.80	40.15	32	38.45	36.28	39.35	38.99	38.50	38.70	38.92	38.74	38.53
		QDF	10.33	30.16	21.53	29.53	24.47	28.95	31.77	32.90	31.39	32.78	31.85	33.16
		LNDF	4.20	31.37	9.77	23.95	11.67	19.76	14.82	16.70	15.12	16.52	15.43	16.37
0	3	LDF	24.13	33.20	29.93	32.37	32.48	34.92	32.72	38.02	38.23	38.78	38.76	39.52
		QDF	14.60	27.01	18.73	25.33	24.40	26.21	28.75	29.77	29.60	30.48	30.57	31.57
		LNDF	2.87	29.47	5.57	19.21	8.20	14.80	10.82	12.29	11.13	12.23	11.24	12.06
1		LDF	24.73	30.65	29.57	30.68	32.58	31.92	37.02	35.99	37.97	36.73	38.95	37.81
		QDF	10.93	23.10	14.90	20.89	20.63	22.29	26.11	24.58	27.03	25.15	27.94	26.15
		LNDF	3.53	26.73	5.50	16.72	7.63	12.68	9.76	9.94	10.31	9.97	10.24	9.80
2		LDF	24	32.02	29.43	31.52	31.95	33.31	37.18	37.57	37.55	38.49	38.62	39.62
		QDF	7.80	22.58	11.73	19.66	17.15	20.62	21.85	22.85	21.05	22.43	22.60	23.50
		LNDF	2.60	24.39	4.13	15.84	6.57	11.73	7.69	9.50	7.57	9.50	7.90	9.37
0	8	LDF	22.07	28.35	31.30	32.28	36.50	37.87	42.43	42.42	43.69	43.54	45.06	45.11
		QDF	3.33	13.43	5.37	10.60	7.18	10.43	9.57	7.94	10.04	7.95	10.39	11.89
		LNDF	.33	13.64	.57	4.61	.78	2.26	1.10	1.20	1.19	1.18	1.28	1.07
1		LDF	24.60	27.23	31.63	31.02	37.55	35.95	42.70	41.88	43.66	43.06	45.04	44.61
		QDF	3.67	13.54	5.83	10.58	6.42	10.50	10.69	11.21	11.01	11.40	11.68	11.78
		LNDF	.27	13.07	.50	4.82	.63	2.10	1.27	1.01	1.13	.97	1.21	1.15
2		LDF	23.73	26.86	31.93	32.22	37.22	36.35	42.60	43.55	43.89	45.05	44.79	44.45
		QDF	2.73	11.68	4.70	8.93	5.90	8.15	8.21	8.63	8.47	9	8.91	9.06
		LNDF	.40	11.17	.50	4.08	.87	2.20	1.08	1.60	.96	1.62	1.04	1.60

Table 3.3: Expected Apparent and Actual error rates (in %) of LDF, QDF and LNDF for simulated skewed data for $p = 10, \delta = .5$ under Skewed vs Skewed population setting.

$p=2$			$\delta=.5$											
			Sample Size											
a	σ^2	$D.F.$	25		50		100		400		600		1000	
			APE	AE	APE	AE	APE	AE	APE	AE	APE	AE	APE	AE
0	1.5	LDF	31.86	33.03	32.43	33.70	35.90	36.72	37.48	37.65	37.71	38.03	37.73	38.09
		QDF	24.46	24.65	23.63	25.03	24.78	26.07	25.65	26.89	26.01	27	25.81	27.20
		LNDF	11	11.82	10.10	11.76	9.43	11.71	10.46	11.64	10.45	11.67	10.47	11.75
1		LDF	33.20	32.70	33.90	32.62	35.91	35.09	36.99	36.57	38.25	37.42	38.08	37.62
		QDF	13.53	11.40	14	11.62	15.03	11.95	15.36	12.59	15.46	12.87	15.22	12.86
		LNDF	8.06	8.07	7.43	7.66	8.31	7.69	8.36	7.55	8.31	7.05	8.11	7.29
2		LDF	30.53	32.75	32.63	34.10	35.33	35.77	37.23	36.95	37.85	38.03	38.23	38.08
		QDF	4.33	4.77	4.76	4.94	5.53	5.05	5.85	5.26	5.91	5.33	6.11	5.36
		LNDF	3.46	3.81	3.76	4.43	4.53	4.32	4.48	4.70	3.57	3.62	4.21	4.34
0	3	LDF	36.20	37.59	39.83	40.92	40.68	41.63	43.69	44.45	43.54	44.38	44.39	44.78
		QDF	22.26	23.27	22.96	23.99	23.21	24.18	23.57	24.57	23.22	24.55	23.96	24.68
		LNDF	7.46	8.33	6.53	7.22	7.53	7.13	7.97	7.17	7.87	7.33	8.11	7.27
1		LDF	35.66	36.55	38.90	38.89	40.75	41.88	43.47	43.65	44.18	44.79	43.90	44.49
		QDF	12.53	14.50	14.73	15.05	15.16	15.43	16.23	15.85	16.28	16	16.30	15.91
		LNDF	5.53	6.79	6.13	6.50	5.76	6.30	7.05	6.60	6.97	6.68	7.33	6.82
2		LDF	36.73	37.26	39.06	38.28	40.63	40.42	43.03	43.21	43.86	43.80	43.78	43.88
		QDF	8.20	6.37	9.13	6.42	9.23	6.65	9.15	6.87	9.22	6.95	9	6.98
		LNDF	3.93	4.28	4.63	3.93	5.03	3.79	4.43	3.42	4.95	4	4.51	3.71
0	8	LDF	39.60	40.60	42.26	43.52	44.88	45.79	47.32	47.73	47.47	47.85	48.26	48.36
		QDF	20.40	19.62	18.96	20.24	19.96	20.36	20.45	20.92	20.23	20.88	20.66	21.10
		LNDF	3.33	4.55	4.63	5.06	4.73	4.41	4.63	4.37	4.37	4.04	4.88	4.29
1		LDF	40.40	41.44	43.66	43.82	44.71	45.69	47.22	47.69	47.56	48	48.09	48.38
		QDF	14.86	15.09	15.56	15.45	15.10	15.67	15.78	15.89	16.17	15.92	16.40	15.95
		LNDF	4	5.02	3.83	4.72	4.08	5.02	4.43	5.04	4.26	5.08	4.14	4.88
2		LDF	40.06	42.47	43.46	44.77	45.73	46.59	47.20	48.10	47.60	48.34	47.93	48.67
		QDF	10.60	12.76	10.80	13	11.05	13.30	11.59	13.65	11.54	13.72	11.88	13.87
		LNDF	3.46	4.70	3.16	4.21	3.10	4.07	3.62	4.26	4.03	4.56	3.50	3.89

Table 3.4: Expected Apparent and Actual error rates (in %) of LDF, QDF and LNDF for simulated skewed data for $p=2, \delta=.5$ under Normal vs Skewed population setting.

$p=2$			$\delta=.5$											
			Skewed vs Skewed						Normal vs Skewed					
a	σ^2	$D.F.$	Sample Size											
			25	50	100	400	600	1000	25	50	100	400	600	1000
0	1.5	LDF	0.52	0.53	0.54	0.54	0.54	0.54	.67	.66	.63	.62	.61	.61
		QDF	0.52	0.53	0.52	0.52	0.52	0.52	.75	.74	.73	.73	.72	.72
		LNDF	0.54	0.55	0.55	0.56	0.56	0.56	.87	.88	.88	.88	.88	.88
1		LDF	0.60	0.59	0.58	0.58	0.58	0.58	.68	.67	.64	.63	.62	.62
		QDF	0.61	0.62	0.60	0.58	.58	0.57	.88	.88	.88	.87	.87	.87
		LNDF	0.67	0.68	0.69	0.70	0.70	0.70	.91	92	.92	.92	.92	.92
2		LDF	0.61	0.58	0.57	0.58	0.58	0.58	.67	.65	.64	.63	.61	.61
		QDF	0.75	0.71	0.68	0.69	0.67	0.69	.95	.95	.94	.94	.94	.94
		LNDF	0.81	0.82	0.83	0.83	0.83	0.83	.96	.95	.95	.95	.96	.95
0	3	LDF	0.56	0.56	.56	0.54	0.54	0.54	.62	.59	.58	.56	.55	.55
		QDF	0.57	0.57	0.57	0.55	0.55	0.55	.76	.76	.75	.75	.75	.75
		LNDF	0.66	0.69	0.69	0.70	0.69	0.69	.91	.92	.92	.92	.92	.92
1		LDF	0.61	0.58	0.57	0.55	0.55	0.55	.63	.61	.58	.56	.55	.55
		QDF	0.66	0.65	0.65	0.63	0.62	0.61	.85	.84	.84	.84	.83	.84
		LNDF	0.70	0.71	0.72	0.72	0.72	0.72	.93	.93	.93	.93	.93	.93
2		LDF	0.59	0.59	0.58	0.56	0.56	0.55	.62	.61	.59	.56	.56	.56
		QDF	0.75	0.75	0.73	0.71	0.71	0.70	.93	.93	.93	.93	.93	.93
		LNDF	0.81	0.82	0.82	0.82	0.82	0.82	.95	.96	.96	.96	.96	.96
0	8	LDF	0.58	0.55	0.54	0.51	0.52	0.51	.56	.56	.54	.52	.52	.51
		QDF	0.68	0.67	0.66	0.65	0.65	0.64	.80	.79	.79	.79	.79	.78
		LNDF	0.81	0.82	0.82	0.82	0.82	0.83	.95	.94	.95	.95	.95	.95
1		LDF	0.58	0.56	0.54	0.52	0.51	0.51	.58	.56	.54	.52	.51	.51
		QDF	0.72	0.71	0.71	0.70	0.69	0.69	.84	.84	.84	.84	.84	.84
		LNDF	0.82	0.83	0.84	0.84	0.84	0.84	.94	.95	.94	.94	.94	.95
2		LDF	0.58	0.55	0.53	0.52	0.51	0.51	.57	.55	.53	.51	.51	.51
		QDF	0.76	0.76	0.76	0.75	0.75	0.74	.87	.86	.86	.86	.86	.86
		LNDF	0.83	0.84	0.84	0.84	0.84	0.84	.95	.95	.95	.95	.95	.95

Table 3.5: Average AC1 statistic of LDF, QDF and LNDF for simulated skewed data with $(p=2, \delta=.5)$.

4

A New Transformation for Normalizing Skewed Data in Classification Problems

4.1 Introduction

As discussed in Chapter 3, moderate deviations of feature data from normality does not seriously hamper the classification accuracies of the normal theory based discriminant functions. However, when one concludes that the deviations from normality of the data under consideration are of serious nature and the normal model would not be an adequate fit, one can still make a wise choice to employ the theoretically robust parametric discriminant functions for classification tasks via suitable data transformations (McLachlan, 2004). Additionally, even the non-parametric methods like LDF can suffer as much as the non-parametric methods when the normality assumption is not met. Hence, wisely chosen data transformations can prove to be effective in achieving approximate normality in datasets and hence consequently can improve the performance of parametric as well as non-parametric discriminant functions over non-normal datasets. The disadvantages of employing transformations in contrast to their potential advantages do not seem compelling in the age of modern computing (Osborne, 2010). In the present work we suggest a multivariate transformation that works on removing skewness from the data and hence aids in improving the performance of the MLC. This chapter is further divided into five Sections, the remaining part of this section discusses the motivation and the objective behind the study conducted in this chapter, Section

4.2 discusses some studies which focused on the issue of employing normality transformations while using MLC, Section 4.3 gives a detailed description of the transformation suggested in the present chapter, Section 4.4 illustrates the relative significance of the suggested transformation in classification tasks as compared to the lognormally transformed data (using two parameter lognormal transform), and untransformed data using an extensive simulation study and a real image dataset as well, and Section 4.5 reports the conclusion of the study.

4.1.1 Motivation

In the previous chapter, a methodology and a new discriminant function based on two parameter lognormal distribution was proposed. The methodology was shown to be robust enough in classifying severely positively skewed data efficiently. However, the assumption of two parameter lognormal class conditional densities used there in is capable of modeling positive skewness and that too for non-zero positive valued observations only. Although, the occurrence of negative observations is not an issue in the field of digital imaging but occurrence of zero valued spectral observations is not a very rare sight in the field. And in such situations, assumption of lognormal class conditional densities would not prove to be a feasible one. Despite this limitation, results obtained in the last chapter indicate that lognormality assumption can be a crucial tool in modeling severe skewness in datasets. Thus, in an attempt to be able to use lognormal distribution for modeling severe skewness of data distributed over the whole range of real line, we make use of the less popular 3− parameter lognormal distribution. We propose and explore a normalizing transformation based on the 3 parameter multivariate Log-normal distribution and exploit it for improving the performance of the Maximum Likelihood Classifier under non-optimal situations of skewness. Although transformations can be crucial tools, they should be employed thoughtfully as they alter the natural form of the data. And hence, for post classification analysis, the data should be reversely transformed for reporting the descriptives.

4.2 Background and Scope of the Study

Data transformations are mathematical modifications that are applied to the data. These mathematical functions can serve many purposes in quantitative

data analysis. In classification problems, the appropriate data transforms can be used as a viable pre processing tool for improving the normality and homogeneity of the populations. Many potential data transformations for improving the normality of data have been proposed so far, with Box-Cox transformation (Box and Cox, 1964) still being the most frequently used ones. Box and Cox (1964) proposed these transformations for univariate scenario only, which require to be applied independently on each of the feature in the datasets thereby ensuring univariate normal distributions of the individual features. But the marginal normal distributions do not certainly ensure the joint multivariate normality which essentially must be the case in multivariate real life classification problems. Later, McLachlan (2004) proposed the multivariate extension of Box-Cox power transformations. Apart from this, square root transformation and log transformation which eventually fall into the class of power transformations or Box-Cox transformations are other two frequently used transforms in the field of classification. The most critical issue with most of these transformations proposed so far is that these are incapable of handling negative values of the observed variables and hence can not cater to such data. Yeo and Johnson (2000) proposed a power transformation which can be defined over the whole real line but defined and illustrated their significance only for univaraite situations. The need to transform the data if found to be non-normal, to improve the performance of discriminant functions, was emphasized upon in many researches (Clarke et al. (1979), Lachenbruch et al. (1973), Beauchamp et al. (1980), McLachlan (2004)). But still in the applied research, pre-processing testing of the data for normality and further consideration for an appropriate normality transformation is mostly ignored. And hence, very few researches utilizing the calibre of normality transforms for improving performance of MLC in classification tasks have been produced so far.

Beauchamp et al. (1980) and Beauchamp and Robson (1986) investigated and illustrated the significance of using appropriate transformations in discriminant analysis and specifically the misclassification probabilities of LDF while using power transformations for acheiving approximate normality and equal covariances in the underlying populations in bivariate situations. Riani and Atkinson (2001) proposed a unified approach for the detection of outliers and illustrated the significance of unifying it along with multivariate transformations in improving the performances of LDF and QDF in the presence of outliers. Ujiie et al. (2002) proposed a modified QDF based on the Box-Cox

transformation with exponential power and showed its effectiveness in reducing misclassification errors for some real datasets but limited their study to the uni-variate distributions only which rarely find application in classification problems. Also, the distributions used by them for displaying the effectiveness of the transformation in reducing the skewness of the data were all positively skewed and hence the performance of the transformation on negatively skewed data is not certain. The square root transformation has often been used for the problem of document classification or handwritten character recognition (Hein and Bousquet (2005), Howard and Jebara (2007), Wakabayashi et al. (1993)) and is preferred for transforming the variables which can be measured as counts as the observations obtained by counting elements generally follow normal distribution (Ujiie et al., 2002).

Parsons et al. (2007) employed the generalized logarithm transformation as a pre-processing variance stabilizing tool and boasted of achieving very high classification accuracy as a result. Logarithms of gene expression ratio were taken for classifying genes from microarray data in Brown et al. (1999). Infact log transform is the most hailed transform among the researchers to deal with skewed data. (Bartlett, 1947; Quenouille, 2014). Elliot (1971) suggested using the lognormal transformation on heavily skewed data in order to reduce the asymmetry of the data. Hoyle (1973) discusses logarithmic transformations as a way of making the data confirm to assumptions of additivity, homogeneity and normality. But both the square root transform as well as the log transforms are defined only on positive observation. For example, a square root transformation or a lognormal transformation can not be defined for the left half values of the real line. Also the logarithmic transformations are found to be unable to handle negatively skewed data which can often be the case with real datasets. Although shifted power transformations have been suggested in order to overcome such issues of negative or zero valued observations in datasets (Atkinson, 1985) but still choosing an appropriate power for using power transformations still remains an issue with the analyst as discussed in Yeo and Johnson (2000). Hence, in the present study a more flexible logarithmic transformation based on the three parameter lognormal distribution has been suggested which is shown to be capable of efficiently handling negatively skewed as well as negative valued observations. Aitchison and Brown (1963); Johnson et al. (1994) and Crow and Shimizu (1988) are the three fundamental texts which provide an excellent insight into the history, theorteical development, parameter estimation, generalizations and detailed applications of the

lognormal distributions.

4.3 3-Parameter Lognormal Distribution and Suggested Transformations

As discussed in Section 4.2, several authors have suggested using simple logarithmic transforms for restoring normality in skewed data. Although level of skewness reduces after applying logarithmic transformations on a skewed dataset but these transformations often do not reduce the skewness to insignificant levels. In such situations the transformed data does not confirm to the approximate normality. Hence, keeping in mind the incapability of logarithmic transforms in handling negatively skewed as well as negative valued observations, we suggest using 3-parameter logarithmic transforms instead, which can efficiently restore normality in negatively skewed data and also is capable of handling negative valued observations. Before defining the the simple logarithmic transform and the suggested transforms, we define first the generalized 3-parameter lognormal distribution and the theory of maximum likelihood estimation of its parameters which happens to be the basis for both of these transformations.

4.3.1 3-parameter lognormal distribution

Aitchison and Brown (1963) define a random variable X as lognormally distributed with mean μ and variance σ^2 if $\ln(X-\tau)$ follows $N(\mu,\sigma^2)$ and denoted X as following $\Lambda(\tau,\mu,\sigma^2)$ with density function defined as

$$g(x)=\frac{1}{(2\pi)^{1/2}\sigma(x-\tau)}\exp\{-(\ln(x-\tau)-\mu)^2/2\},\qquad \tau<x<\infty \tag{4.1}$$

where, τ with range $\tau<x$ defines the lower bound for the distribution of X and is therefore referred to as the threshold parameter or the shift parameter. And hence the two parameter lognormal distribution is defined as a special case of the 3- parameter lognormal distribution when $\tau=0$ with density function defined as

$$g(x)=\frac{1}{(2\pi)^{1/2}\sigma x}\exp\{-(\ln x-\mu)^2/2\},\qquad x>0. \tag{4.2}$$

As for negative values of X, $\ln(X-\tau)$ can still be defined for $|\tau| > |X|$, hence the three parameter lognormal distribution can efficiently handle negative valued observations as well.

4.3.2 Negatively skewed lognormal distribution

The lognormal distributions defined above can be used for modeling positively skewed distributions only but Aitchison and Brown (1963) discussed the modifications in the above definition of lognormal distribution which are capable of modeling negatively skewed distributions as well . They defined X as following negatively skewed lognormal distribution with mean μ, variance σ^2 and threshold parameter τ if $\ln(\tau - X)$ follows $N(\mu, \sigma^2)$ where $-\infty < X < \tau$,with pdf,

$$g(x) = \frac{1}{(2\pi)^{1/2}\sigma(\theta - x)} \exp\{-(\ln(\theta - x) - \mu)^2/2\}, \qquad -\infty < x < \theta. \quad (4.3)$$

The above described densities in equations (4.1) and (4.3) are of univariate positively and negatively skewed lognormal distributions respectively but as the classification problems generally deal with multivariate data, we define the multivariate counterparts of the above densities.
For a p-dimensional random vector $(X_1, X_2, \ldots, X_p)'$ of positive random variables such that $(\ln X_1, \ln X_2, \ldots \ln X_p)'$ follows a p-dimensional normal distribution with mean vector $\boldsymbol{\mu} = (\mu_1, \mu_2, \ldots, \mu_p)'$ and a positive definite variance covariance matrix $\boldsymbol{\Sigma} = \{(\sigma_{ij}), i, j = 1, 2, \ldots, p\}$, Crow and Shimizu (1988) defined $(X_1, X_2, \ldots X_p)'$ as following a 2-parameter p-variate lognormal distribution $\Lambda_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with density function defined as

$$g(\mathbf{x}) = \frac{1}{(2\pi)^{p/2} \mid \boldsymbol{\Sigma} \mid^{1/2} x_1.x_2 \ldots x_p} \exp\left[-\frac{(\ln \mathbf{x} - \boldsymbol{\mu})' \boldsymbol{\Sigma}^{-1} (\ln \mathbf{x} - \boldsymbol{\mu})}{2}\right] \quad (4.4)$$

where, $\mathbf{x} = (x_1, x_2, \ldots, x_p)'$, $x_j > 0, j = 1, 2, \ldots, p$.

Now, we define a random vector $(X_1, X_2, \ldots, X_p)'$ such that $(\ln(X_1 - \tau_1)$, $\ln(X_2 - \tau_2) \ldots, \ln(X_p - \tau_p))'$ follows a p-dimensional normal distribution with mean vector $\boldsymbol{\mu} = (\mu_1, \mu_2, \ldots, \mu_p)'$ and a positive definite variance-covariance matrix $\boldsymbol{\Sigma} = \{(\sigma_{ij}), i, j = 1, 2, \ldots, p\}$. Then the vector $(X_1, X_2, \ldots, X_p)'$ has a 3- parameter p-variate positively skewed lognormal distribution $\Lambda_p(\boldsymbol{\tau}, \boldsymbol{\mu}, \boldsymbol{\Sigma})$

with pdf,

$$g\left(\mathbf{x}\right)=\frac{1}{(2\pi)^{p/2}|\ \boldsymbol{\Sigma}\ |^{1/2}\prod_{j=1}^{p}(x_k-\tau_k)}\exp\left[-\frac{\left(\ln\left(\mathbf{x}-\boldsymbol{\tau}\right)-\boldsymbol{\mu}\right)'\boldsymbol{\Sigma}^{-1}\left(\ln\left(\mathbf{x}-\boldsymbol{\tau}\right)-\boldsymbol{\mu}\right)}{2}\right] \tag{4.5}$$

where, $\mathbf{x}=(x_1,x_2,\ldots,x_p)'$, and $\boldsymbol{\tau}=(\tau_1,\tau_2,\ldots\tau_p)'$ is the threshold parameter with $\tau_j<x_j<\infty, j=1,2,\ldots,p$.
And a random vector $(X_1,X_2,\ldots,X_p)'$ such that $(\ln(\theta_1-X_1),\ln(\theta_2-X_2),\ldots,\ln(\theta_p-X_p))'$ follows a p-dimensional normal distribution with mean vector $\boldsymbol{\mu}=(\mu_1,\mu_2,\ldots,\mu_p)'$ and a positive definite variance covariance matrix $\boldsymbol{\Sigma}=\{(\sigma_{ij}),i,j=1,2,\ldots,p\}$, has a 3- parameter p-varaite negatively skewed lognormal distribution $\Lambda_p(\boldsymbol{\theta},\boldsymbol{\mu},\boldsymbol{\Sigma})$ with pdf,

$$g\left(\mathbf{x}\right)=\frac{1}{(2\pi)^{p/2}|\ \boldsymbol{\Sigma}\ |^{1/2}\prod_{j=1}^{p}(\theta_j-x_j)}\exp\left[-\frac{\left(\ln\left(\boldsymbol{\theta}-\mathbf{X}\right)-\boldsymbol{\mu}\right)'\boldsymbol{\Sigma}^{-1}\left(\ln\left(\boldsymbol{\theta}-\mathbf{x}\right)-\boldsymbol{\mu}\right)}{2}\right] \tag{4.6}$$

where, $\mathbf{x}=(x_1,x_2,\ldots,x_p)'$, and $\boldsymbol{\theta}=(\theta_1,\theta_2,\ldots\theta_p)'$ is the threshold parameter of the negatively skewed lognormal distribution with $-\infty<x_j<\theta_j, j=1,2,\ldots,p$.

4.3.3 Maximum likelihood estimation of $\boldsymbol{\tau},\boldsymbol{\theta},\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$

The involvement of the threshold parameter complicates the parameter estimation procedure for the 3−parameter lognormal distribution. Crow and Shimizu (1988) discussed various estimation procedures for the parameter estimation of generalizd lognormal distributions and their limitaions. It is clear from their discussion that global maximum likelihood estimates and the moment estimates can lead to inadmissible results in the parameter estimation procedure for the generalized lognormal distribution. However, the local maximum liklelihood estimates (LMLE) obtained by equating the partial derivatives of the loglikelihhod to zero appear to give reasonable estimates in most of the situations and also possess the desirable properties of maximum likelihood estimates as well. Thus, we obtain the expressions for LMLE's of the parameters of multivariate 3−parameter lognormal distribution which will be used for defining a new set of normalizing transformations further.

If $\mathbf{x} = \{x_{ij}, i = 1, 2, \ldots, n, j = 1, 2, \ldots, p\}$ is a random sample from $A_p(\boldsymbol{\tau}, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ then using the density function defined in equation (4.5) the log-likelihood function can be defined as

$$\begin{aligned} \ln L(\boldsymbol{\tau}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) =& \frac{-np}{2} \ln 2\pi - \frac{n}{2} \ln |\boldsymbol{\Sigma}| - \sum_{i=1}^{n} \sum_{j=1}^{p} \ln (x_{ij} - \tau_j) \\ & - \frac{1}{2} \sum_{i=1}^{n} \left[(\ln(\mathbf{x}_i - \boldsymbol{\tau}) - \boldsymbol{\mu})' \boldsymbol{\Sigma}^{-1} (\ln(\mathbf{x}_i - \boldsymbol{\tau}) - \boldsymbol{\mu}) \right] \end{aligned} \tag{4.7}$$

where, $\mathbf{x}_i$ is the $(p \times 1)$ vector of observations for the ith sample from the population .

Now equating the partial derivatives, of the above given loglikelihood with respect to $\boldsymbol{\mu}, \boldsymbol{\tau}$ and $\boldsymbol{\Sigma}$ respectively, equal to zero we obtain the following LMLE estimating equations.

$$\begin{aligned} \frac{\partial \ln L}{\partial \boldsymbol{\mu}} &= \sum_{i=1}^{n} \left[\boldsymbol{\Sigma}^{-1} (\ln(\mathbf{x}_i - \boldsymbol{\tau}) - \boldsymbol{\mu}) \right] \\ &= 0 \qquad (4.8) \\ \frac{\partial \ln L}{\partial \boldsymbol{\Sigma}} &= -\frac{1}{2} \left(\boldsymbol{\Sigma}^{-1} \right) - \frac{1}{2} \sum_{i=1}^{n} \left[-\boldsymbol{\Sigma}^{-1} (\ln(\mathbf{x}_i - \boldsymbol{\tau}) - \boldsymbol{\mu}) (\ln(\mathbf{x}_i - \boldsymbol{\tau}) - \boldsymbol{\mu})' \boldsymbol{\Sigma}^{-1} \right] \\ &= 0 \qquad (4.9) \\ \frac{\partial \ln L}{\partial \boldsymbol{\tau}} &= \sum_{i=1}^{n} \frac{1}{(\mathbf{x}_i - \boldsymbol{\tau})} + \sum_{i=1}^{n} \left[diag(\mathbf{x}_i - \boldsymbol{\tau})^{-1} \boldsymbol{\Sigma}^{-1} (\ln(\mathbf{x}_i - \boldsymbol{\tau}) - \boldsymbol{\mu}) \right] \\ &= 0 \qquad (4.10) \end{aligned}$$

Simplifying equations (4.8) and (4.9), we get the following LML estimates of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$:

$$\hat{\boldsymbol{\mu}} = \frac{1}{n} \sum_{i=1}^{n} \ln \left(\mathbf{x}_i - \hat{\boldsymbol{\tau}} \right) \tag{4.11}$$

$$\hat{\boldsymbol{\Sigma}} = \frac{1}{n} \sum_{i=1}^{n} \left(\ln(\mathbf{x}_i - \hat{\boldsymbol{\tau}}) - \hat{\boldsymbol{\mu}} \right) \left(\ln(\mathbf{x}_i - \hat{\boldsymbol{\tau}}) - \hat{\boldsymbol{\mu}} \right)' \tag{4.12}$$

Substituting $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ in equation (4.10) with their MLE's from the above set of equations, the resulting implicit equation in $\boldsymbol{\tau}$ becomes

$$\begin{aligned} f(\hat{\boldsymbol{\tau}}) &= \sum_{i=1}^{n} \frac{1}{(\mathbf{x}_i - \hat{\boldsymbol{\tau}})} + \sum_{i}^{n} \Big[diag(\mathbf{x}_i - \hat{\boldsymbol{\tau}})^{-1} \hat{\boldsymbol{\Sigma}}^{-1} \big(\ln(\mathbf{x}_i - \hat{\boldsymbol{\tau}}) - \hat{\boldsymbol{\mu}} \big) \Big] \\ &= 0 \end{aligned} \tag{4.13}$$

where, $diag(\mathbf{x}_i - \hat{\boldsymbol{\tau}})^{-1}$ is a $(p \times p)$ matrix with $(\mathbf{x}_i - \hat{\boldsymbol{\tau}})^{-1}$ as its main diagonal. This equation can be solved iteratively for obtaining an unbiased estimate of the threshold parameter $\boldsymbol{\tau}$ and consequetively of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$.

On the similar lines the LMLE's of $\boldsymbol{\tau}, \boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ for negatively skewed 3−parameter lognormal distribution can be derived and are expressed as:

$$\hat{\boldsymbol{\mu}} = \frac{1}{n} \sum_{i}^{n} \ln \big(\hat{\boldsymbol{\theta}} - \mathbf{x}_i \big) \tag{4.14}$$

$$\hat{\boldsymbol{\Sigma}} = \frac{1}{n} \sum_{i}^{n} \big(\hat{\boldsymbol{\theta}} - \mathbf{x}_i) - \hat{\boldsymbol{\mu}} \big) \big(\hat{\boldsymbol{\theta}} - \mathbf{x}_i) - \hat{\boldsymbol{\mu}} \big)' \tag{4.15}$$

$$\begin{aligned} f(\hat{\boldsymbol{\theta}}) &= \sum_{i}^{n} \frac{1}{(\hat{\boldsymbol{\theta}} - \mathbf{x}_i)} + \sum_{i}^{n} \Big[diag(\hat{\boldsymbol{\theta}} - \mathbf{x}_i)^{-1} \hat{\boldsymbol{\Sigma}}^{-1} \big(\ln(\hat{\boldsymbol{\theta}} - \mathbf{x}_i) - \hat{\boldsymbol{\mu}} \big) \Big] \\ &= 0 \end{aligned} \tag{4.16}$$

4.3.4 Suggested transformations

Now, assuming three parameter lognormal distributions for the underlying skewed populations in a classification problem, we define a new set of transformations. The transformations for positively skewed data have been denoted as T_{PS} and the ones for negatively skewed data as T_{NS}.

We define T_{PS} as the transform for normalizing an $(n \times p)$-dimensional positively skewed data matrix $\mathbf{X} = \{x_{ij}, i = 1, 2 \ldots, n; j = 1, 2, \ldots, p\}$ from a population as,

$$T_{PS} : \ln (\mathbf{X} - \mathbf{T}) \tag{4.17}$$

where, $\mathbf{T} = \{\tau_{ij}, i =, 2, \ldots, n; j = 1, 2, \ldots, p\}$ is an $(n \times p)$ dimensional matrix with each $\boldsymbol{\tau}_i = (\tau_1, \tau_2, \ldots \tau_p)'$ representing the threshold parameter for the population. The vector $\boldsymbol{\tau}_i$, is obtained from the sample training data using equation (4.13) as the maximum likelihood estimate of the threshold parameter assuming 3- parameter multivariate positively skewed lognormal distribution for the underlying positively skewed population. Let us mention here that if

the underlying populations confirm to a two parameter multivariate lognormal distribution then the estimated values of $\boldsymbol{\tau}_i$ should come out as an approximate zero vector and hence T_{PS} reduces to the simple logarithmic transformation, $ln(\mathbf{X})$.

And T_{NS} is defined as the transform for normalizing an $(n \times p)$-dimensional negatively skewed training data matrix $\mathbf{X} = \{x_{ij}, i = 1, 2 \ldots, n; j = 1, 2, \ldots, p\}$ from a population as

$$T_{NS} : \ln(\mathbf{T} - \mathbf{X}) \tag{4.18}$$

where, $\mathbf{T}$ is an $n \times p$ dimensional matrix with $\boldsymbol{\theta}_i = (\theta_1, \theta_2, \ldots \theta_p)'$ as each of its n rows which is to be estimated from the sample training data using equation (4.16) as the maximum likelihood estimate of the threshold parameter assuming three parameter multivariate negatively skewed lognormal distribution for the underlying negatively skewed population .
The threshold parameter required for applying the suggested transformations was estimated using R software (Team, 2013) and the classification was performed in MATLAB (Matlab, 2013b)

4.4 Numerical Illustration

4.4.1 Data generation via simulation

In order to illustrate the capability of the suggested transform in acting as a catalyst for improving the performance of the MLC by normalizing negatively as well as positively skewed data efficiently, we designed a simulation study based on tri-variate two class problem. The training datasets from both the populations were simulated from two parameter lognormal distributions first in R software using package *compositions* and then the three parameter lognormally distributed populations were generated from these populations by introducing a vector $\mathbf{t}$. The first trivariate population was kept as fixed and was simulated from standard lognormal distribution with parameters $\mu_1 = (0, 0, 0)'$ and $\Sigma_1 = I_3$ while the second population was simulated repeatedly for various combinations of the different values of the mean vector, the variance covariance matrix and the training sample size given in Table 4.1. If $\mathbf{X} = \{x_{ij}, i = 1, 2 \ldots, n; j = 1, 2, \ldots, p\}$ is the $(n \times p)$ training data matrix then adding $\mathbf{t} = (1, 2, 3)'$, to each row of $\mathbf{X}$, the simulated populations were

transformed so as to further induce significant positive skewness in the data. Similarly applying operation $\mathbf{t} - \mathbf{X}$, to each row of $\mathbf{X}$, the simulated datasets were transformed to generate negatively skewed datasets. Also, separate index samples of size 1000 were generated from both the populations which were used for validating the classifier.

After successfully generating positively as well as negatively skewed training datasets, we assumed these simulated datasets to be following 3−parameter lognormal distribution and used the *optim* function in R for finding the maximum likelihood estimate of the threshold parameter $\boldsymbol{\tau}$ from the training datasets. After estimating the vector $\boldsymbol{\tau}$ for each of the training datasets, the appropriate transformations defined in (4.17) and (4.18) were used for normalizing the training as well as index datasets and the QDF was used for classification, afterwards. In order to illustrate the relative improvement in the classification of skewed data via suggested transformations, the performance of LDF and QDF on untransformed simulated datasets and of QDF on simply log-transformed datasets have also been measure in terms of error rates. For better understanding of the analysis results, we rename the QDF based on the suggested transformations as the *3LogTr* classifier and the QDF based on simple logarithmic transformation as the *LogTr* classifier. The whole process of simulating training datasets, estimating the threshold parameter, training the MLC using, transformed training datasets in case of *3LogTr* and *LogTr* whereas, using untransformed training datasets in case of LDF and QDF and classifying the index sample using the trained classifier was repeated 30 times for each of the 4 classifiers. The average classification accuracies for the training datasets which are termed as Apparent Accuracies (APAc) and the average classification accuracies for index samples which are termed as Actual Accuracies (AAc) for positively and negatively skewed datasets were calculated over 30 replications for each of the 4 classifiers namely LDF, QDF, 3LogTr, LogTr and have been tabulated in Tables 4.4 and 4.5 respectively. To exhibit the efficiency of the transformations in significantly reducing the skewness, the skewness of the untransformed simulated datasets as well as of transformed datasets was calculated for the second population of all the positively as well as negatively skewed simulated datsets for $n = 100$ and $n = 25$ respectively which are reported in Table 4.3.

Apart from finding the overall accuracies (APAc and AAc) for the 4 classifiers, the *Area under the ROC curves* (AUROC) and the Gwet's AC1 statistic were also calculated for assessing the classification performance of all the 4 classi-

fiers which are tabulated in Tables 4.6 and 4.7. Based on the results tabulated in Tables 4.4, 4.5, 4.6 and 4.7, the performances of the 4 classifiers in terms of AAc and AUROC were plotted for a few simulated datasets which are shown in Figures 4.1, 4.2, 4.3 and 4.4.

Table 4.1: Values of data characteristics used for simulating second population.

Mean Vector of second population	$\boldsymbol{\mu_{21}} = (0.5, 0, 0)$ $\boldsymbol{\mu_{22}} = (0.5, .5, 0)$ $\boldsymbol{\mu_{23}} = (0.5, 0.5, 0.5)$ $\boldsymbol{\mu_{24}} = (0.5, 1, 2)$ $\boldsymbol{\mu_{25}} = (1, 2, 3)$
Covariance Matrix of second population	$\boldsymbol{\Sigma_2} = \sigma^2 I_3$ $\sigma^2 = (1.5, 3, 8)$
Size of Training sample from each class	$n = (25, 50, 100, 1000)$

4.4.2 Results

An extensive simulation study was performed in this study with an objective to check the performance of the suggested set of transformations based on the 3−parameter lognormal distribution in restoring normality in skewed data and consequently in improving the performance of MLC. The performance of the MLC on untransformed data, logarithmically transformed data and on the data transformed using suggested transformations was compared using simulated datasets. All the three classification performance checks via overall accuracies, area under ROC curves and Gwet's AC1 statistic respectively signify the better performance of the MLC based on the suggested transformations (named as 3LogTr classifier in analysis) over LDF, QDF and the MLC based on the simple log transformation for almost all the positively as well as negatively skewed simulated datasets. Apart from this, the following findings with respect to different data characteristics considered in the study were observed for the simulated data.

- *Effect of transformation on skewness:* The results reported in the Table 4.3 clearly signify the ability of the suggested transformations in reducing the skewness of significantly skewed datasets to a larger extent as compared to the simple logarithmic transformations. Apart from calculating the multivariate skewness, we also tested the transformed datasets for significant skewness using Mardia's test (Mardia, 1974) and found all

the datasets to be having insignificant skewness when transformed using suggested transformations. Whereas, when the datasets were transformed using simple logarithmic transformations, Mardia's test reported the presence of significant skewness. It implies that the suggested transformations have greater capability of restoring normality in the skewed datasets. The logarithmic transformations failed on negatively skewed data as it contains negative valued observations also and hence only skewness coefficients for log-transformed positively skewed data have been repoted in the Table 4.3.

- *Effect of transformations on classification accuracies:* The suggested transformations significantly reduced the skewness in the datasets. This confirms the normality of datasets, and hence the suggested 3LogTr classifier resulted in higher classification accuracies as compared to LDF, QDF and LogTr classifiers in every case as is evident from the Tables 4.4 and 4.5 and the Figures 4.1, 4.2, 4.3 and 4.4. Moreover, it was observed that the 3LogTr classifier depicted distinct improvement in the classification performance even when the separability between the two populations is quite less i.e. when $\mu_2 = \mu_{21} = (.5, 0, 0)$ (see Figures 4.1, 4.2, 4.3 and 4.4). However, as the separability between the two populations increases with larger values of the mean vector of the second population (μ_2), the performances of all the 4 classifiers was found to be improving with 3LogTr performing exceptionally well with AUROC values as high as .99.

- *Effect of skewness:* It is evident from the APAc, AAc and the respective plots constructed in this study that as the levels of skewness increased in the datasets with the increased value of σ^2, the performance of QDF was found to be deteriorating further, whereas, the performance of TrQDF continued to improve in contrast.

- The inability of general logarithmic transformations in accounting for the negative skewness in data is clearly reflected from the performance of the LogTr classifier given in the Table 4.5 and the Figures 4.3, 4.4. Accuracy Plots in the Figures 4.3 and 4.4 and a 0.5 value of the AUROC in the Table 4.7 show that the performance of QDF on negatively skewed data transformed using simple logarithmic transformations results in random classification of the observations.

- *Effect of training data size:* The 3LogTr classifier performed comparatively better than the other three classifiers for almost all the training data sizes and it should also be noted that the classifier based on the suggested transformations was found to be empirically robust for small training data sizes (i.e. $n < 30$) as well, that can be observed from the Figures 4.1, 4.2, 4.3 and 4.4.

- *Random agreement measure:* The measure of random agreement in classifier's performance calculated in this study via Gwet's AC1 statistic was found to be lying in the range $(.6, .8)$ for the suggested transformation based MLC (i.e. 3LogTr). This implies that the 3LogTr classifier results in fair to excellent levels of agreeement.

4.4.3 Application

To illustrate the performance of the suggested transformations on real life datasets, we employed the SPOT dataset described in Section 2.4.2 which consisits of 8 landuse categories. A record of the various classes and their size is given in Table 4.2. The performance of the LDF, QDF and the MLC based on log transformations on the SPOT dataset has already been assessed in Section 2.4.3.2 and the misclassification errors are reported in Table 2.2. For performing classification via the suggested transformation based classifier 3LogTr, the dataset was divided into mutually exclusive training and testing datasets. 50 random samples from each class were randomly selected and kept in testing dataset and the remaining samples were used for training the classifier. As the training data of all the 8 landuse categories was found to be significantly skewed (2.4.2), the suggested transformations were used for transforming the training and the testing datasets of all the 8 categories. The values of the threshold parameters estimated from the training data of the 6 categories are given in Table 4.2. The misclassification error of the 3LogTr classifier on the testing dataset was found to be 17.27% which reports a significant decrease as compared to the misclassification errors obtained using LDF, QDF (see, Table 2.2) and LogTr (see, Table 3.1).

4.5 Conclusion

The new set of normalizing transformations suggested in this chapter were found to be effective as well as flexible enough in significantly reducing the

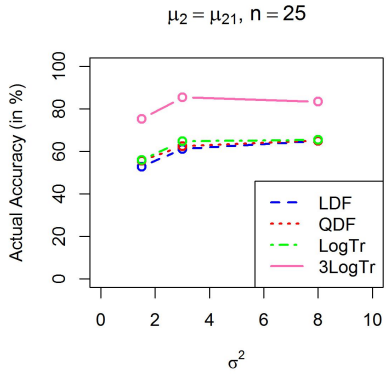
Landuse categories	Size of each class	Type of skewness	$\boldsymbol{\tau}_i$ for each class
Class 1	8425	Positive skewness	$(0, 112.86)$
Class 2	347	Not skewed	$(0, 0)$
Class 3	105	Not skewed	$(0, 0)$
Class 4	276	Positively skewed	(69.24,0)
Class 5	288	Not skewed	$(0, 0)$
Class 6	157	Positively skewed	135.24
Class 7	273	Not skewed	$(0, 0)$
Class 8	129	Positively skewed	$(-64.16, 116.37)$

Table 4.2: Estimated threshold parameters for the significantly skewed training datasets of SPOT data.

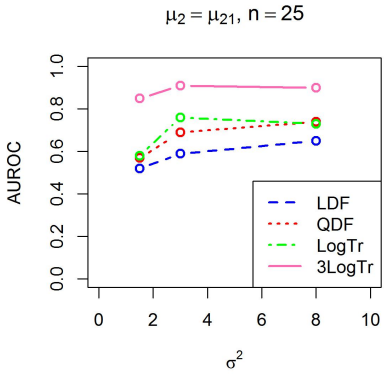
skewness of any nature (i.e. either positive or negative skewness) present in data. The results of the simulation study exhibit the calibre of the suggested transformations in restoring normality in positively as well as negatively skewed datasets. The contribution of the suggested transformations in improving the performance of the MLC has also been justified through simulation study as well as through real dataset. In the light of the results obtained in this chapter, we conclude that the suggested transformations may act as an aid in improving the performance of MLC and hence should be considered as a pre-processing aid while classifying skewed data. We understand the challenges involved in estimating the parameters of a generalized lognormal distribution and intend to conduct further research for simplifying the procedures.

		Skewness of				
σ^2	$\boldsymbol{\mu_2}$	Positively skewed data			Negatively skewed data	
		Untransformed data	3LogTr	LogTr	Untransformed data	3LogTr
1.5	$\boldsymbol{\mu}_{21}$	40.38	0.26	7.40	11.33	.97
	$\boldsymbol{\mu}_{22}$	33.51	0.47	6.62	21.35	1.30
	$\boldsymbol{\mu}_{23}$	24.58	0.58	3.99	15.88	.44
	$\boldsymbol{\mu}_{24}$	66.53	0.09	6.31	17.01	1.67
	$\boldsymbol{\mu}_{25}$	76.91	0.27	3.64	6	1.47
3	$\boldsymbol{\mu}_{21}$	112.81	0.39	9.86	26.08	.90
	$\boldsymbol{\mu}_{22}$	51.19	0.33	6.90	27.95	3.50
	$\boldsymbol{\mu}_{23}$	59.06	0.79	9.87	35.08	.93
	$\boldsymbol{\mu}_{24}$	71.84	0.24	5.81	7.95	1.62
	$\boldsymbol{\mu}_{25}$	48.79	0.69	2.98	25.94	1.09
3	$\boldsymbol{\mu}_{21}$	34.41	0.44	6.08	53.60	1.91
	$\boldsymbol{\mu}_{22}$	97.78	0.42	8.77	34.33	2.39
	$\boldsymbol{\mu}_{23}$	156.49	0.46	11.58	23.93	1.59
	$\boldsymbol{\mu}_{24}$	130.35	0.49	6.58	23.91	1.59
	$\boldsymbol{\mu}_{25}$	42.42	1.00	2.77	23.91	1.59

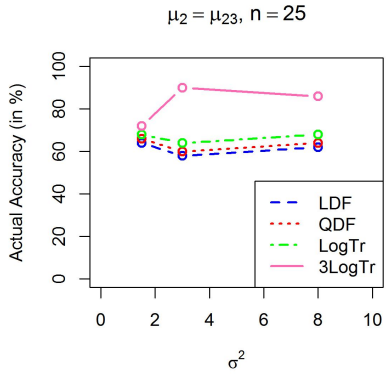
Table 4.3: Mardia's multivariate coefficient of skewness for second population of simulated positively and negatively skewed datasets.



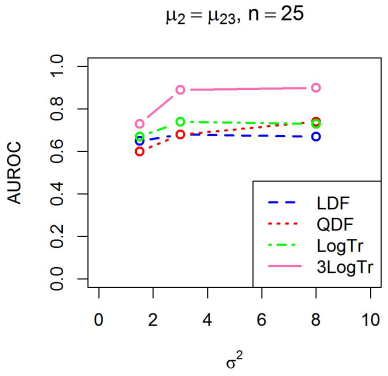
(a) *Accuracy plot*



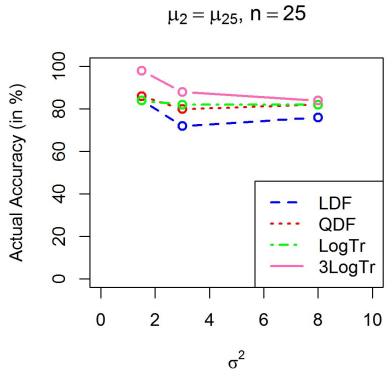
(b) *AUROC plot*



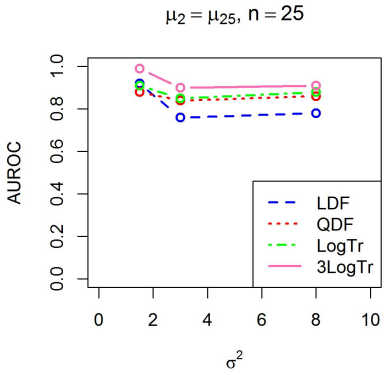
(c) *Accuracy plot*



(d) *AUROC plot*

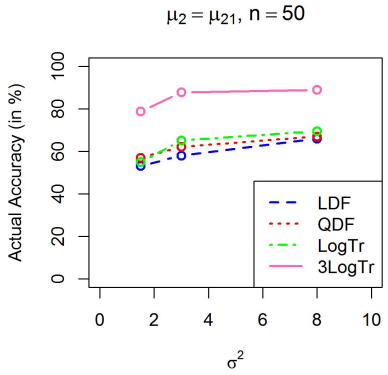


(e) *Accuracy plot*

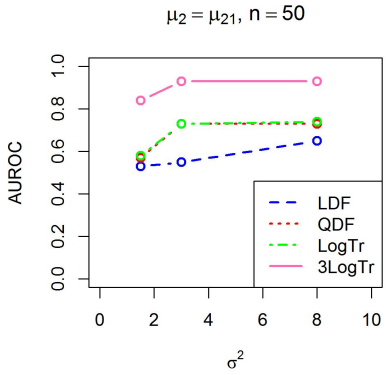


(f) *AUROC plot*

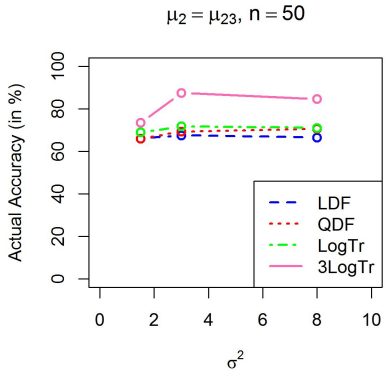
Figure 4.1: Plots of expected actual accuracies of LDF, QDF, QDF on log-transformed data (LogTr) and QDF on 3−parameter log-transformed data over positively skewed simulated index sample for $n = 25$, and $\mu_2 = (\mu_{21}, \mu_{23}$ and $\mu_{25})$ depicting the effect of variability on accuracies.



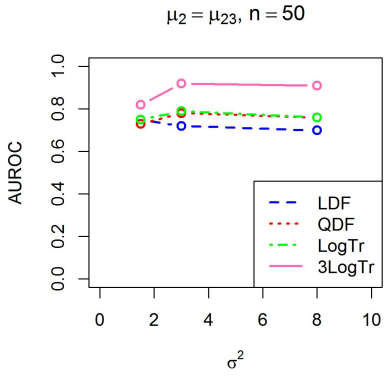
(a) *Accuracy plot*



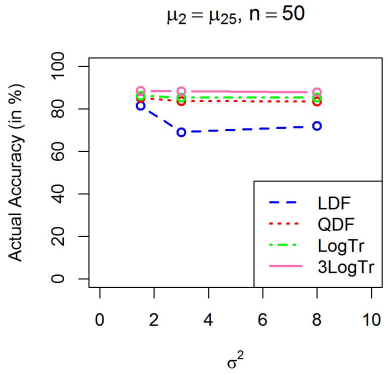
(b) *AUROC plot*



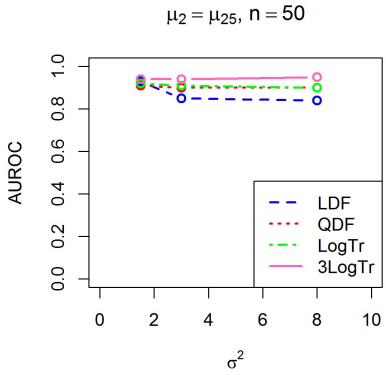
(c) *Accuracy plot*



(d) *AUROC plot*

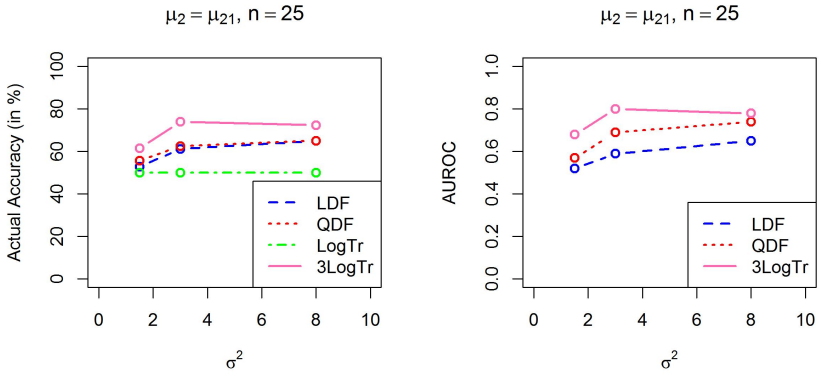


(e) *Accuracy plot*



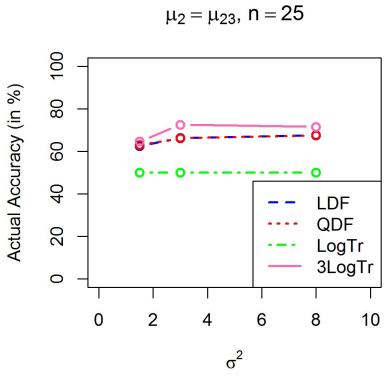
(f) *AUROC plot*

Figure 4.2: Plots of expected actual accuracies of LDF, QDF, QDF on log-transformed data (LogTr) and QDF on 3−parameter log-transformed data over positively skewed simulated index sample for $n = 50$, and $\mu_2 = (\mu_{21}, \mu_{23}$ and $\mu_{25})$ depicting the effect of variability on accuracies.

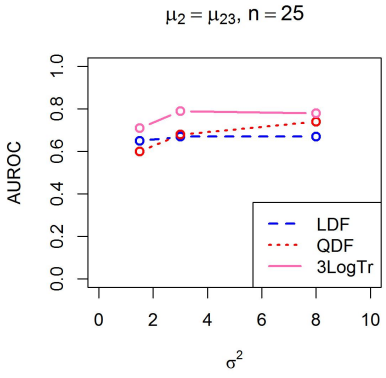


(a) *Accuracy plot*

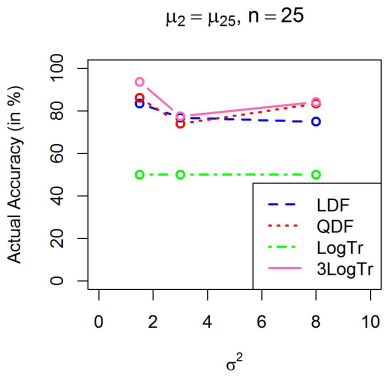
(b) *AUROC plot*



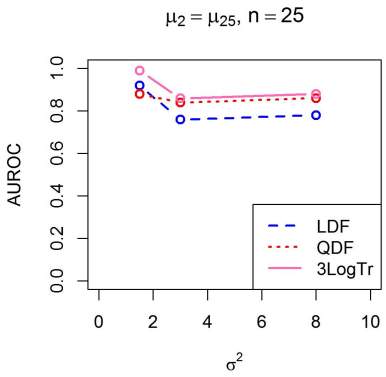
(c) *Accuracy plot*



(d) *AUROC plot*

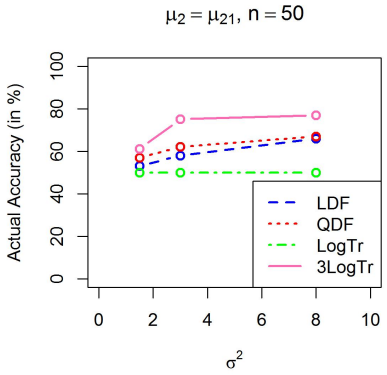


(e) *Accuracy plot*

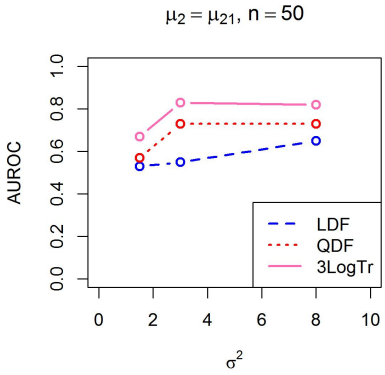


(f) *AUROC plot*

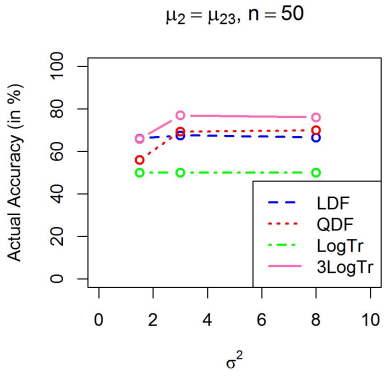
Figure 4.3: Plots of expected actual accuracies of LDF, QDF, QDF on log-transformed data (LogTr) and QDF on 3−parameter log-transformed data over negatively skewed simulated index sample for $n = 25$, and $\mu_2 = (\mu_{21}, \mu_{23}$ and $\mu_{25})$ depicting the effect of variability on performance of LDF and QDF on transformed and untransformed data.



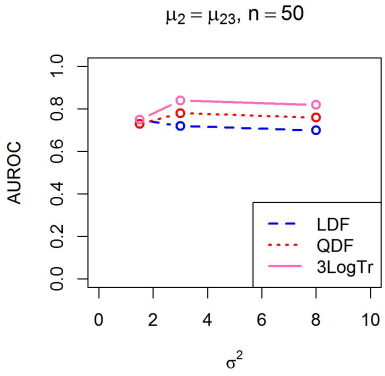
(a) *Accuracy plot*



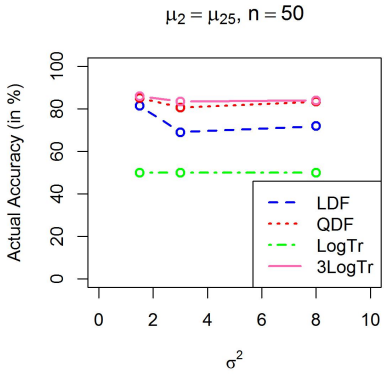
(b) *AUROC plot*



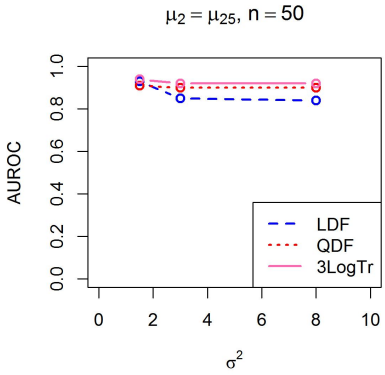
(c) *Accuracy plot*



(d) *AUROC plot*



(e) *Accuracy plot*



(f) *AUROC plot*

Figure 4.4: Plots of expected actual accuracies of LDF, QDF, QDF on log-transformed data (LogTr) and QDF on 3−parameter log-transformed data over negatively skewed simulated index sample for $n = 50$, and $\mu_2 = (\mu_{21}, \mu_{23}$ and $\mu_{25})$ depicting the effect of variability on performance of LDF and QDF on transformed and untransformed data.

$\boldsymbol{\mu}$	σ^2	*D.F.*	Sample Size							
			25		50		100		1000	
			APAc	AAc	APAc	AAc	APAc	AAc	APAc	AAc
1.5	$\boldsymbol{\mu}_{21}$	LDF	48	52.88	59	53.25	57.50	54.50	56.35	55.50
		QDF	56	55.75	58	57.13	56.50	55.13	56.0	54.63
		LogTr	58	56.0	60	55.13	56.0	54.75	57.55	55.87
		3LogTr	**86.0**	**75.38**	**78.0**	**78.87**	**78.0**	**81.0**	**80.70**	**80.87**
	$\boldsymbol{\mu}_{22}$	LDF	50	53.50	57	51.25	64.50	59.62	59.15	61.50
		QDF	56.0	54.0	60.0	54.75	61.0	57.25	57.45	57.63
		LogTr	58.0	54.13	58	54.50	63.50	60.0	61.60	61.88
		3LogTr	**90**	**76.88**	**82.0**	**76.88**	**65.50**	**66.50**	**62.85**	**62.88**
	$\boldsymbol{\mu}_{23}$	LDF	64	62.25	79	66.13	64.50	68.37	67.45	66.50
		QDF	66.0	63.12	78.0	66.13	61.50	65.13	63.20	62.75
		LogTr	68.0	63.75	81.0	69.0	65.50	67.50	68.70	68.0
		3LogTr	**72**	**69.75**	**83.0**	**73.50**	**71.0**	**77.75**	**70.10**	**68.63**
	$\boldsymbol{\mu}_{24}$	LDF	68	64.62	71	66.63	66	66.13	66	65.50
		QDF	66.0	66.63	68.0	64.38	65.50	65.25	61.50	63.88
		LogTr	66.00	64.75	74.0	66.37	67.50	67.0	67.70	67.63
		3LogTr	**76.0**	**72.12**	**77.0**	**74.50**	**68.50**	**67.87**	**69.40**	**68.75**
	$\boldsymbol{\mu}_{25}$	LDF	84	83.50	74	81.50	72.50	69.50	75.30	80.75
		QDF	86.0	86.25	78	85.25	83.0	83.50	76.35	78.75
		LogTr	84.0	85.88	81.0	86.25	89.0	86.0	83.95	86.75
		3LogTr	**98.0**	**94.13**	**84.0**	**88.50**	**91.0**	**85.25**	**86.0**	**86.75**
3	$\boldsymbol{\mu}_{21}$	LDF	54	61.25	63	58.13	67	62.75	63.40	63.88
		QDF	60.0	62.50	70.0	62.25	66.50	63.38	61.75	62.0
		LogTr	62.0	64.88	71.0	65.25	67.50	65.50	65.30	65.75
		3LogTr	**86.0**	**85.50**	**90.0**	**87.88**	**74.0**	**69.50**	**73.45**	**73.12**
	$\boldsymbol{\mu}_{22}$	LDF	52	53.37	70	64	67.50	66.87	64.65	66
		QDF	64.0	61.12	70.0	66.37	66.0	67.63	64.45	65.63
		LogTr	68.0	61.88	73.0	68.0	65.50	68.63	68.45	68.75
		3LogTr	**92.0**	**82.67**	**83.0**	**87.38**	**70.50**	**71.13**	**73.95**	**74.38**
	$\boldsymbol{\mu}_{23}$	LDF	58	66.37	70	67.63	64	67.37	67.25	68.37
		QDF	60.0	66.25	72.0	69.37	69.0	69.75	67.65	69.25
		LogTr	64.0	69.13	71.0	71.88	69.0	72.0	70.50	71.88
		3LogTr	**90.0**	**83.0**	**85.0**	**87.62**	**73.50**	**74.25**	**74.15**	**76.25**
	$\boldsymbol{\mu}_{24}$	LDF	60	74.50	76	71.13	70.50	71.63	69.60	69.87
		QDF	58.0	72.0	74.0	73.62	69.50	72.25	69.10	70.0
		LogTr	66.0	74.38	78.0	75.0	72.50	74.62	73.50	74.88
		3LogTr	**80.0**	**72.12**	**89.0**	**87.0**	**76.0**	**74.88**	**76.65**	**76.12**
	$\boldsymbol{\mu}_{25}$	LDF	72	76.88	68	69.13	78.50	79.13	71.90	70.75
		QDF	80.0	84.0	84.0	83.75	85.0	83.88	79.95	78.63
		LogTr	82.0	84.13	86.0	85.38	87.0	86.0	86.0	85.62
		3LogTr	**88.0**	**82.25**	**87.0**	**88.38**	**89.50**	**84.13**	**87.35**	**86.0**
8	$\boldsymbol{\mu}_{21}$	LDF	58	65.13	68	66	65	66.25	62.95	65.87
		QDF	62.0	65.13	69.0	67.13	64.50	67.63	61.45	64.85
		LogTr	68.0	65.50	77.0	69.63	64.50	69.0	66.30	69.75
		3LogTr	**90.0**	**83.50**	**89**	**89.0**	**77.50**	**72.38**	**72.10**	**75.38**
	$\boldsymbol{\mu}_{22}$	LDF	60	62.62	68	64.62	68.50	65.87	63.10	63.62
		QDF	64.0	62.75	67.0	66.87	67.50	67	63.20	66.37
		LogTr	64.0	63.62	72.0	68.0	68.50	68.50	67.30	68.63
		3LogTr	**78**	**81.50**	**86.0**	**85.62**	**71.50**	**71.75**	**71.85**	**72.36**
	$\boldsymbol{\mu}_{23}$	LDF	62	67.75	63	66.63	64.50	69	67.15	68.25
		QDF	64.0	67.50	73.0	70.75	68.0	68.50	68.20	68.0
		LogTr	68.0	68.25	79.0	71.13	69.0	71.13	71.70	71.0
		3LogTr	**86.0**	**81.87**	**86.0**	**84.75**	**74.50**	**72.12**	**76.40**	**74.38**
	$\boldsymbol{\mu}_{24}$	LDF	66	58.63	74	70.37	68.50	68.37	69.25	68.87
		QDF	72.0	67.13	77.0	73.25	70.50	71.50	68.70	69.50
		LogTr	74.0	67.75	81.0	73.88	74.50	72.88	73.75	72.75
		3LogTr	**90.0**	**82.0**	**83.0**	**86.25**	**79.0**	**73.62**	**76.20**	**75.38**
	$\boldsymbol{\mu}_{25}$	LDF	76	75	71	72	76	74.12	70.70	70.87
		QDF	82.0	83.50	84.0	83.50	86.0	84.0	79.30	78.37
		LogTr	82.0	84.62	85.0	85.38	88.0	85.25	83.95	84.75
		3LogTr	**84.0**	**87.38**	**88.0**	**87.88**	**87.50**	**83.63**	**85.80**	**85.75**

Table 4.4: Expected Apparent Accuracies (APAcs) and Expected Actual Accuracies (AAcs) (in %) of LDF, QDF, LogTr and 3LogTr for simulated positively skewed data.

$\boldsymbol{\mu}$	σ^2	*D.F.*	Sample Size							
			25		50		100		1000	
			APAc	AAc	APAc	AAc	APAc	AAc	APAc	AAc
1.5	$\boldsymbol{\mu}_{21}$	LDF	48	52.88	59	53.25	57.50	54.50	56.35	55.50
		QDF	56	55.75	58	57.13	56.50	55.13	56	54.63
		LNDF	48	50	49	50	49.50	50	49.95	50
		TrQDF	**70**	**61.62**	**65**	**61.25**	**58.50**	**59.50**	**61.70**	**60**
	$\boldsymbol{\mu}_{22}$	LDF	50	53.50	57	51.25	64.50	59.62	59.95	61.50
		QDF	56	54	60	54.75	61	57.25	57.45	57.63
		LNDF	48	50	49	50	49.50	50	49.95	50
		TrQDF	**64**	**64.50**	**68**	**60.88**	**65.50**	**60.50**	**62.85**	**63**
	$\boldsymbol{\mu}_{23}$	LDF	64	62.55	79	66.13	64.50	68.37	67.45	66.50
		QDF	66.0	63.12	78.0	56.13	61.50	56.13	63.20	62.75
		LNDF	48	50	49	50	49.50	50	49.95	50
		TrQDF	**66**	**64.75**	**80**	**65.87**	**69**	**68**	**70.05**	**68.63**
	$\boldsymbol{\mu}_{24}$	LDF	68	64.62	71	66.63	66	66.13	66	65.50
		QDF	66.0	66.63	68.0	64.38	65.50	65.25	61.50	63.88
		LNDF	48	50	49	50	49.50	50	49.95	50
		TrQDF	**74**	**68.38**	**75**	**66.25**	**68.50**	**67.87**	**69.50**	**68.87**
	$\boldsymbol{\mu}_{25}$	LDF	84	83.50	74	81.50	72.50	69.50	75.30	80.75
		QDF	86.0	86.25	78	85.25	83.0	83.50	76.35	78.75
		LNDF	48	50	49	50	49.50	50	49.95	50
		TrQDF	**98.0**	**93.75**	**83.0**	**86.12**	**91.0**	**85.25**	**86.0**	**86.75**
3	$\boldsymbol{\mu}_{21}$	LDF	54	61.25	63	58.13	67	62.75	63.40	63.88
		QDF	60.0	62.50	70.0	62.25	66.50	63.38	61.75	62.0
		LNDF	48	50	49	50	49.50	50	49.95	50
		TrQDF	**78.0**	**74.0**	**80.0**	**75.25**	**74.0**	**69.63**	**73.45**	**73.12**
	$\boldsymbol{\mu}_{22}$	LDF	52	53.37	70	64	67.50	66.87	64.65	66
		QDF	64.0	61.12	70.0	66.37	66.0	67.63	64.45	65.63
		LNDF	48	50	49	50	49.50	50	49.95	50
		TrQDF	**78**	**71.88**	**80.0**	**76.62**	**70.50**	**71.25**	**73.95**	**74.38**
	$\boldsymbol{\mu}_{23}$	LDF	58	66.37	70	67.63	64	67.37	67.25	68.37
		QDF	60.0	66.25	72.0	69.37	69.0	69.75	67.65	69.25
		LNDF	48	50	49	50	49.50	50	49.95	50
		TrQDF	**78**	**72.62**	**81.0**	**77**	**73.50**	**74.25**	**74.15**	**76.12**
	$\boldsymbol{\mu}_{24}$	LDF	60	74.50	76	71.13	70.50	71.63	69.60	69.87
		QDF	58.0	72.0	74.0	73.62	69.50	72.25	69.10	70.0
		LNDF	48	50	49	50	49.50	50	49.95	50
		TrQDF	**80.0**	**76.13**	**87.0**	**78.37**	**76.0**	**74.88**	**76.70**	**76.12**
	$\boldsymbol{\mu}_{25}$	LDF	72	76.88	68	69.13	78.50	79.13	71.90	70.75
		QDF	80	74	84	80.75	85	80.88	79.95	78.63
		LNDF	48	50	49	50	49.50	50	49.95	50
		TrQDF	**82.0**	**77.50**	**90**	**83.50**	**90**	**84.25**	**87.35**	**86.0**
8	$\boldsymbol{\mu}_{21}$	LDF	58	65.13	68	66	65	66.25	62.95	65.87
		QDF	62.0	65.13	69.0	67.13	64.50	67.63	61.45	64.88
		LNDF	48	50	49	50	49.50	50	49.95	50
		TrQDF	**78**	**72.38**	**81**	**77**	**77.50**	**72.50**	**72.10**	**75.38**
	$\boldsymbol{\mu}_{22}$	LDF	60	62.62	68	64.62	68.50	65.87	63.10	63.62
		QDF	64.0	62.75	67.0	66.87	67.50	67	63.20	66.37
		LNDF	48	50	49	50	49.50	50	49.95	50
		TrQDF	**70**	**70.75**	**81.0**	**75.75**	**71.50**	**71.75**	**71.85**	**72.38**
	$\boldsymbol{\mu}_{23}$	LDF	62	67.75	63	66.63	64.50	69	67.15	68.25
		QDF	64.0	67.50	73.0	70	68.0	68.50	68.20	68.0
		LNDF	48	50	49	50	49.50	50	49.95	50
		TrQDF	**76.0**	**71.63**	**82.0**	**76.12**	**74.50**	**72.25**	**76.40**	**74.25**
	$\boldsymbol{\mu}_{24}$	LDF	66	58.63	74	70.37	68.50	68.37	69.25	68.87
		QDF	72.0	67.13	77.0	70.25	70.50	71.50	68.70	69.50
		LNDF	48	50	49	50	49.50	50	49.95	50
		TrQDF	**76**	**72.88**	**86.0**	**74.75**	**79.0**	**73.75**	**76.20**	**75.38**
	$\boldsymbol{\mu}_{25}$	LDF	76	75	71	72	76	74.12	75	75.13
		QDF	82.0	83.50	84.0	83.50	86.0	84.0	79.30	78.37
		LNDF	48	50	49	50	49.50	50	49.95	50
		TrQDF	**88.0**	**84.25**	**88.0**	**84.13**	**87.50**	**83.63**	**85.80**	**85.75**

Table 4.5: Expected Apparent Accuracies (APAcs) and Expected Actual Accuracies (AAcs) (in %) of LDF, QDF, LogTr and 3LogTr for simulated positively skewed data.

$\boldsymbol{\mu}$	σ^2	*D.F.*	AUC				AC1 statistic			
			Sample Size							
			25	50	100	1000	25	50	100	1000
1.5	$\boldsymbol{\mu}_{21}$	LDF	.52	.53	.54	.54	.52	.53	.54	.55
		QDF	.57	.57	.58	.60	.55	.57	.55	.54
		LogTr	.58	.58	.57	.59	.55	.55	.54	.55
		TrQDF	**.85**	**.84**	**.66**	**.61**	**.75**	**.79**	**.54**	**.59**
	$\boldsymbol{\mu}_{22}$	LDF	.51	.45	.61	.62	.53	.51	.59	.61
		QDF	.58	.53	.59	.64	.53	.54	.57	.57
		LogTr	.61	.54	.60	.64	.53	.54	.59	.61
		TrQDF	**.81**	**.81**	**.62**	**.66**	**.76**	**.76**	**.60**	**.62**
	$\boldsymbol{\mu}_{23}$	LDF	.65	.75	.76	.75	.62	.66	.68	.66
		QDF	.60	.73	.72	.72	.63	.66	.65	.62
		LogTr	.67	.75	.74	.75	.63	.68	.67	.67
		TrQDF	**.73**	**.82**	**.85**	**.76**	**.69**	**.73**	**.77**	**.68**
	$\boldsymbol{\mu}_{24}$	LDF	.70	.71	.71	.72	.64	.66	.66	.65
		QDF	.68	.63	.71	.72	.64	.64	.65	.63
		LogTr	.68	.63	.71	.71	.66	.66	.66	.67
		TrQDF	**.78**	**.82**	**.74**	**.73**	**.72**	**.74**	**.67**	**.68**
	$\boldsymbol{\mu}_{25}$	LDF	.92	.93	.81	.91	.83	.81	.69	.80
		QDF	.88	.91	.91	.90	.86	.85	.83	.78
		LogTr	.91	.92	.93	.93	.85	.86	.85	.86
		TrQDF	**.99**	**.94**	**.94**	**.93**	**.94**	**.85**	**.85**	**.86**
3	$\boldsymbol{\mu}_{21}$	LDF	.59	.55	.60	.61	.61	.58	.62	.63
		QDF	.69	.73	.73	.74	.62	.62	.63	.61
		LogTr	.76	.73	.74	.74	.64	.65	.65	.65
		TrQDF	**.91**	**.93**	**.86**	**.80**	**.85**	**.87**	**.69**	**.73**
	$\boldsymbol{\mu}_{22}$	LDF	.45	.63	.67	.67	.53	.63	.66	.65
		QDF	.59	.73	.77	.77	.61	.66	.67	.65
		LogTr	.63	.73	.76	.77	.61	.67	.68	.68
		TrQDF	**.89**	**.93**	**.88**	**.81**	**.82**	**.87**	**.71**	**.74**
	$\boldsymbol{\mu}_{23}$	LDF	.68	.72	.71	.74	.66	.67	.67	.68
		QDF	.68	.78	.79	.80	.66	.69	.69	.69
		LogTr	.74	.79	.80	.81	.69	.71	.71	.71
		TrQDF	**.89**	**.92**	**.86**	**.84**	**.82**	**.87**	**.74**	**.76**
	$\boldsymbol{\mu}_{24}$	LDF	.77	.76	.77	.77	.74	.71	.71	.69
		QDF	.73	.80	.81	.81	.71	.73	.72	.69
		LogTr	.79	.81	.81	.81	.74	.74	.74	.74
		TrQDF	**.79**	**.92**	**.86**	**.83**	**.72**	**.86**	**.74**	**.76**
	$\boldsymbol{\mu}_{25}$	LDF	.76	.85	.90	.87	.76	.69	.79	.70
		QDF	.84	.90	.91	.91	.83	.83	.83	.78
		LogTr	.85	.91	.92	.92	.84	.85	.85	.85
		TrQDF	**.90**	**.94**	**.93**	**.92**	**.82**	**.88**	**.84**	**.85**
8	$\boldsymbol{\mu}_{21}$	LDF	.65	.65	.65	.65	.65	.65	.66	.65
		QDF	.74	.73	.75	.76	.65	.67	.67	.76
		LogTr	.73	.74	.76	.76	.65	.69	.68	.76
		TrQDF	**.90**	**.93**	**.78**	**.81**	**.83**	**.88**	**.72**	**.81**
	$\boldsymbol{\mu}_{22}$	LDF	.64	.64	.68	.64	.62	.64	.65	.63
		QDF	.72	.73	.75	.76	.62	.66	.66	.66
		LogTr	.69	.74	.75	.76	.63	.67	.68	.68
		TrQDF	**.90**	**.91**	**.77**	**.78**	**.81**	**.85**	**.71**	**.72**
	$\boldsymbol{\mu}_{23}$	LDF	.67	.70	.72	.73	.67	.66	.68	.68
		QDF	.74	.76	.77	.78	.67	.70	.68	.67
		LogTr	.73	.76	.77	.78	.68	.71	.71	.70
		TrQDF	**.90**	**.91**	**.79**	**.80**	**.81**	**.84**	**.72**	**.74**
	$\boldsymbol{\mu}_{24}$	LDF	.47	.74	.72	.73	.58	.70	.68	.68
		QDF	.68	.78	.79	.80	.67	.73	.71	.69
		LogTr	.73	.78	.79	.79	.67	.73	.72	.72
		TrQDF	**.89**	**.92**	**.79**	**.81**	**.81**	**.86**	**.73**	**.75**
	$\boldsymbol{\mu}_{25}$	LDF	.78	.84	.88	.83	.74	.71	.74	.70
		QDF	.86	.90	.90	.90	.83	.83	.83	.78
		LogTr	.88	.90	.91	.91	.84	.85	.85	.84
		TrQDF	**.91**	**.95**	**.91**	**.92**	**.87**	**.87**	**.83**	**.85**

Table 4.6: Area under ROC curve and Gwet's AC1 statistic values of LDF, QDF, LogTr and 3LogTr classifiers for simulated positively skewed data

$\boldsymbol{\mu}$	σ^2	*D.F.*	AUC				AC1 statistic			
			Sample Size							
			25	50	100	1000	25	50	100	1000
1.5	$\boldsymbol{\mu}_{21}$	LDF	.52	.53	.54	.54	.52	.53	.54	.55
		QDF	.57	.57	.58	.60	.55	.57	.55	.54
		TrQDF	**.68**	**.67**	**.56**	**.61**	**.61**	**.61**	**.54**	**.59**
	$\boldsymbol{\mu}_{22}$	LDF	.51	.45	.61	.62	.53	.51	.59	.61
		QDF	.58	.53	.59	.64	.53	.54	.57	.57
		TrQDF	**.70**	**.64**	**.62**	**.66**	**.64**	**.60**	**.60**	**.62**
	$\boldsymbol{\mu}_{23}$	LDF	.65	.75	.76	.75	.62	.66	.68	.66
		QDF	.60	.73	.72	.72	.63	.66	.65	.62
		TrQDF	**.71**	**.75**	**.74**	**.76**	**.64**	**.65**	**.67**	**.68**
	$\boldsymbol{\mu}_{24}$	LDF	.70	.71	.71	.72	.64	.66	.66	.65
		QDF	.68	.63	.71	.71	.66	.64	.65	.63
		TrQDF	**.68**	**.74**	**.74**	**.73**	**.63**	**.66**	**.67**	**.68**
	$\boldsymbol{\mu}_{25}$	LDF	.92	.93	.81	.91	.83	.81	.69	.80
		QDF	.88	.91	.91	.90	.86	.85	.83	.78
		TrQDF	**.99**	**.94**	**.93**	**.93**	**.93**	**.86**	**.85**	**.86**
3	$\boldsymbol{\mu}_{21}$	LDF	.59	.55	.60	.61	.61	.58	.62	.63
		QDF	.69	.73	.73	.74	.62	.62	.63	.61
		TrQDF	**.80**	**.83**	**.76**	**.80**	**.73**	**.75**	**.69**	**.73**
	$\boldsymbol{\mu}_{22}$	LDF	.45	.63	.67	.67	.53	.63	.66	.65
		QDF	.59	.73	.77	.77	.61	.66	.67	.65
		TrQDF	**.77**	**.83**	**.78**	**.81**	**.71**	**.76**	**.71**	**.74**
	$\boldsymbol{\mu}_{23}$	LDF	.67	.72	.71	.74	.66	.67	.67	.68
		QDF	.68	.78	.79	.80	.66	.69	.69	.69
		TrQDF	**.79**	**.84**	**.82**	**.84**	**.72**	**.76**	**.74**	**.76**
	$\boldsymbol{\mu}_{24}$	LDF	.77	.76	.77	.77	.74	.71	.71	.69
		QDF	.73	.80	.81	.81	.71	.73	.72	.69
		TrQDF	**.76**	**.86**	**.82**	**.83**	**.70**	**.78**	**.75**	**.76**
	$\boldsymbol{\mu}_{25}$	LDF	.76	.85	.90	.87	.76	.69	.79	.70
		QDF	.84	.90	.91	.91	.83	.83	.83	.78
		TrQDF	**.86**	**.92**	**.93**	**.93**	**.77**	**.83**	**.84**	**.85**
8	$\boldsymbol{\mu}_{21}$	LDF	.65	.65	.65	.65	.65	.65	.66	.65
		QDF	.74	.73	.75	.76	.65	.67	.67	.64
		TrQDF	**.78**	**.82**	**.78**	**.81**	**.72**	**.76**	**.73**	**.75**
	$\boldsymbol{\mu}_{22}$	LDF	.64	.64	.68	.64	.62	.64	.65	.63
		QDF	.72	.73	.75	.76	.62	.66	.66	.66
		TrQDF	**.77**	**.81**	**.75**	**.78**	**.70**	**.75**	**.71**	**.72**
	$\boldsymbol{\mu}_{23}$	LDF	.67	.70	.72	.73	.67	.66	.68	.68
		QDF	.74	.76	.77	.78	.67	.70	.68	.67
		TrQDF	**.78**	**.82**	**.78**	**.80**	**.71**	**.76**	**.72**	**.74**
	$\boldsymbol{\mu}_{24}$	LDF	.47	.74	.72	.73	.58	.70	.68	.68
		QDF	.68	.78	.79	.80	.67	.73	.71	.69
		TrQDF	**.80**	**.81**	**.79**	**.82**	**.72**	**.74**	**.74**	**.75**
	$\boldsymbol{\mu}_{25}$	LDF	.78	.84	.88	.83	.74	.71	.74	.70
		QDF	.86	.90	.90	.90	.83	.83	.83	.78
		TrQDF	**.88**	**.92**	**.91**	**.91**	**.84**	**.84**	**.83**	**.85**

Table 4.7: Area under ROC curve and Gwet's AC1 statistic values of LDF, QDF, LogTr and 3LogTr classifiers for simulated negatively skewed data.

Bibliography

Aeberhard, S., Coomans, D., and De Vel, O. (1994). Comparative analysis of statistical pattern recognition methods in high dimensional settings. *Pattern Recognition*, 27(8):1065–1077.

Aitchison, J. and Brown, J. A. C. (1963). *The Lognormal Distribution*. Cambridge University Press.

Aizerman, M., Braverman, E. M., and Rozonoer, L. (1964). Theoretical foundation of potential function method in pattern recognition. *Automation and Remote Control*, 25:917–936.

Aronoff, S. (1982). Classification accuracy-a user approach. *Photogrammetric Engineering and Remote Sensing*, 48(8):1299–1307.

Aronoff, S. (1985). The minimum accuracy value as an index of classification accuracy. *Photogrammetric Engineering and Remote Sensing*, 51(1):99–111.

Atiya, A. F. (2001). Bankruptcy prediction for credit risk using neural networks: A survey and new results. *Neural Networks, IEEE Transactions on*, 12(4):929–935.

Atkinson, A. C. (1985). *Plots, Transformations, and Regression: An Introduction to Graphical Methods of Diagnostic Regression Analysis*. Clarendon Press.

Atkinson, P. M. (1991). Optimal ground-based sampling for remote sensing investigations: estimating the regional meant. *International Journal of Remote Sensing*, 12(3):559–567.

Atkinson, P. M. (1996). Optimal sampling strategies for raster-based geographical information systems. *Global Ecology and Biogeography Letters*, 5:271–280.

Atkinson, P. M. and Tatnall, A. (1997). Introduction neural networks in remote sensing. *International Journal of Remote Sensing*, 18(4):699–709.

Azzalini, A. and Capitanio, A. (1999). Statistical applications of the multivariate skew normal distribution. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):579–602.

Bache, K. and Lichman, M. (2013). UCI machine learning repository. `http://archive.ics.uci.edu/ml`.

Ball, G. and Hall, D. (1965). Isodata, a novel method of data analysis and pattern classification. *Technical report/Stanford research institute*.

Ball, G. H. and Hall, D. J. (1967). A clustering technique for summarizing multivariate data. *Behavioral science*, 12(2):153–155.

Baron, A. (1991). Misclassification among methods used for multiple group discrimination-the effects of distributional properties. *Statistics In Medicine*, 10:757–766.

Bartlett, M. S. (1947). The use of transformations. *Biometrics*, 3(1):39–52.

Beauchamp, J., Folkert, J., and Robson, D. (1980). A note on effect of logarithmic transformations on the probability of misclassification. *Communications in Statistics*, A9:777–794.

Beauchamp, J. and Robson, D. (1986). Transformation considerations in discriminant analysis. *Communications in Statistics-Simulation and Computation*, 15(1):147–179.

Ben-Hur, A., Horn, D., Siegelmann, H. T., and Vapnik, V. (2002). Support vector clustering. *The Journal of Machine Learning Research*, 2:125–137.

Benediktsson, J. A., Swain, P. H., and Ersoy, O. K. (1990). Neural network approaches versus statistical methods in classification of multisource remote sensing data. *Geoscience and Remote Sensing, IEEE Transactions on*, 28(4):540–552.

Borg, I. and Groenen, P. J. (2005). *Modern Multidimensional Scaling: Theory and Applications*. Springer Science & Business Media.

Boulesteix, A.-L., Strobl, C., Augustin, T., and Daumer, M. (2008). Evaluating microarray-based classifiers: an overview. *Cancer Informatics*, 6.

Box, G. E. and Cox, D. R. (1964). An analysis of transformations. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 211–252.

Breiman, L. (1996a). Bagging predictors. *Machine learning*, 24(2):123–140.

Breiman, L. (1996b). Out-of-bag estimation. Technical report, Citeseer.

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.

Breiman, L., Freidman, J., Olshen, R., and Stone, C. (1984). *Classification and Regression Trees*. Wadsworth and Brooks/Cole, First edition.

Briem, G. J., Benediktsson, J. A., and Sveinsson, J. R. (2002). Multiple classifiers applied to multisource remote sensing data. *Geoscience and Remote Sensing, IEEE Transactions on*, 40(10):2291–2299.

Brodley, C. E. and Friedl, M. A. (1999). Identifying mislabeled training data. *Journal of Artificial Intelligence Research*, 11:131–167.

Brown, M. P., Grundy, W. N., Lin, D., Cristianini, N., Sugnet, C., Ares, M., and Haussler, D. (1999). Support vector machine classification of microarray gene expression data. *University of California, Santa Cruz, Technical Report UCSC-CRL-99-09*.

Bruzzone, L., Cossu, R., and Vernazza, G. (2004). Detection of land-cover transitions by combining multidate classifiers. *Pattern Recognition Letters*, 25(13):1491–1500.

Burges, C. C. (1998). A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167.

Byrt, T., Bishop, J., and Carlin, J. B. (1993). Bias, prevalence and kappa. *Journal of clinical epidemiology*, 46(5):423–429.

Camargo, L. and Yoneyama, T. (2001). Specification of training sets and the number of hidden neurons for multilayer perceptrons. *Neural computation*, 13(12):2673–2680.

Chan, J. C.-W. and Paelinckx, D. (2008). Evaluation of random forest and adaboost tree-based ensemble classification and spectral band selection for ecotope mapping using airborne hyperspectral imagery. *Remote Sensing of Environment*, 112(6):2999–3011.

Chang, C.-C. and Lin, C.-J. (2011). Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27.

Chapelle, O., Vapnik, V., Bousquet, O., and Mukherjee, S. (2002). Choosing multiple parameters for support vector machines. *Machine Learning*, 46(1-3):131–159.

Chaudhuri, P., Ghosh, A., and Oja, H. (2009). Classification based on hybridization of parametric and nonparametric classifiers. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(7):1153–1164.

Clarke, W., Lachenbruch, P., and Broffitt, B. (1979). How non normality affects the quadratic discriminant function. *Communications in statistics*, A8(13):1285–1301.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.

Congalton, R. G. (1988). A comparison of sampling schemes used in generating error matrices for assessing the accuracy of maps generated from remotely sensed data. *Photogrammetric Engineering and Remote Sensing (USA)*, 54:593–600.

Congalton, R. G. (1991). A review of assessing the accuracy of classifications of remotely sensed data. *Remote Sensing of Environment*, 37(1):35–46.

Congalton, R. G. and Green, K. (2008). *Assessing the Accuracy of Remotely Sensed Data: Principles and Practices*. CRC press.

Cortijo, F. and Blanca, D. L. (1997). A comparative study of some nonparametric spectral classifiers: application to problems with high overlapping training sets. *International Journal of Remote Sensing*, 18:1259–1275.

Crawford, M. M., Ham, J., Chen, Y., and Ghosh, J. (2003). Random forests of binary hierarchical classifiers for analysis of hyperspectral data. In *Advances in Techniques for Analysis of Remotely Sensed Data, 2003 IEEE Workshop on*, pages 337–345. IEEE.

Crow, E. and Shimizu, K. (1988). *Lognormal Distributions: Theory and Applications*. M. Dekker, New York.

Curran, P. J. (1988). The semivariogram in remote sensing: an introduction. *Remote sensing of Environment*, 24(3):493–507.

Cutler, D. R., Edwards Jr, T. C., Beard, K. H., Cutler, A., Hess, K. T., Gibson, J., and Lawler, J. J. (2007a). Random forests for classification in ecology. *Ecology*, 88(11):2783–2792.

Cutler, D. R., T.C. Edwards, J., Beard, K., Cutler, A., and et al., K. H. (2007b). On the comparison of classifiers for microarray data. *Ecological Society of America*, 88(11):2783–2792.

Dahl, G. E., Yu, D., Deng, L., and Acero, A. (2012). Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition. *Audio, Speech, and Language Processing, IEEE Transactions on*, 20(1):30–42.

Davis, S. M., Landgrebe, D. A., Phillips, T. L., Swain, P. H., Hoffer, R. M., Lindenlaub, J. C., and Silva, L. F. (1978). Remote sensing: the quantitative approach. *New York, McGraw-Hill International Book Co., 1978. 405 p.*, 1.

DeFries, R. and Chan, J. C.-W. (2000). Multiple criteria for evaluating machine learning algorithms for land cover classification from satellite data. *Remote Sensing of Environment*, 74(3):503–515.

DeLeo, J. M. (1993). The receiver operating characteristic function as a tool for uncertainty management in artificial neural network decision-making. In *Uncertainty Modeling and Analysis, 1993. Proceedings., Second International Symposium on*, pages 141–144. IEEE.

DeLeo, J. M. and Rosenfeld, S. J. (2001). Essential roles for receiver operating characteristic (roc) methodology in classifier neural network applications. In *Neural Networks, 2001. Proceedings. IJCNN'01. International Joint Conference on*, volume 4, pages 2730–2731. IEEE.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, 7:1–30.

Diaz-Uriarte, R. and de Andres, A. (2006). Gene selection and classification of microarray data using random forests. *BMC Bioinformatics*, 7:3.

Dillon, W. R. (1979). The performance of the linear discriminant function in nonoptimal situations and the estimation of classification error rates: A review of recent findings. *Journal of Marketing Research*, pages 370–381.

Dossat, N., Mangé, A., Solassol, J., Jacot, W., Lhermitte, L., Maudelonde, T., Daurès, J.-P., and Molinari, N. (2007). Comparison of supervised classification methods for protein profiling in cancer diagnosis. *Cancer Informatics*, 3.

Dudoit, S., Fridlyand, J., and Speed, T. P. (2002). Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal of the American Statistical Association*, 97(457):77–87.

Elliot, J. (1971). *Some Methods for the Statistical Analysis of Samples of Benthic Invertebrates*. Freshwater Biological Association.

Erbeka, F. S., Ozkan, C., and Taberner, M. (2004). Comparison of maximum likelihood classification method with supervised artificial network algorithms for land use activities. *International Journal of Remote Sensing*, 25(9):1733–1748.

Evans, F. H. (1998). *An Investigation into the use of Maximum Likelihood Classifiers, Decision Trees, Neural Networks and Conditional probabilistic networks for mapping and predicting salinity*. Citeseer.

Fielding, A. H. and Bell, J. F. (1997). A review of methods for the assessment of prediction errors in conservation presence/absence models. *Environmental conservation*, 24(01):38–49.

Fisher, R. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7(2):179–188.

Flach, P. A. (2003). The geometry of roc space: understanding machine learning metrics through roc isometrics. In *ICML*, pages 194–201.

Fleming, M., Berkebile, J., and Hoffer, R. (1975). Computer-aided analysis of landsat-1 mss data: A comparison of three approaches, including a" modified clustering" approach. *LARS Technical Reports*.

Foody, G. M. (1992). On the compensation for chance agreement in image classification accuracy assessment. *Photogrammetric engineering and remote sensing*, 58(10):1459–1460.

Foody, G. M. (1996). Approaches for the production and evaluation of fuzzy land cover classifications from remotely-sensed data. *International Journal of Remote Sensing*, 17(7):1317–1340.

Foody, G. M. (2002). Status of land cover classification accuracy assessment. *Remote sensing of environment*, 80(1):185–201.

Foody, G. M. and Mathur, A. (2004). Towards intelligent training of supervised image classifications: directing training data acquisition for svm classification. *Remote Sensing of Environment*, 93(1):107–117.

Foody, G. M. and Mathur, A. (2006). The use of small training sets containing mixed pixels for accurate hard image classification: Training on mixed spectral responses for classification by a svm. *Remote Sensing of Environment*, 103(2):179–189.

Friedl, M. A. and Brodley, C. E. (1997). Decision tree classification of land cover from remotely sensed data. *Remote Sensing of Environment*, 61(3):399–409.

Friedl, M. A., Sulla-Menashe, D., Tan, B., Schneider, A., Ramankutty, N., Sibley, A., and Huang, X. (2010). Modis collection 5 global land cover: Algorithm refinements and characterization of new datasets. *Remote Sensing of Environment*, 114(1):168–182.

Friedman, J. H. (1989). Regularized discriminant analysis. *Journal of the American Statistical Association*, 84(405):165–175.

Fukuda, T. and Shibata, T. (1992). Theory and applications of neural networks for industrial control systems. *IEEE Transactions on Industrial Electronics*, 39(6):472–489.

Fukunaga, K. (2013). *Introduction to Statistical Pattern Recognition*. Academic press.

Gale, H. (1967). Some examples of the application of the lognormal distribution in radiation protection. *Annals of Occupational Hygiene*, 10(1):39–45.

Gall, J., Razavi, N., and Van Gool, L. (2012). An introduction to random forests for multi-class object detection. In *Outdoor and Large-Scale Real-World Scene Analysis*, pages 243–263. Springer.

Garson, G. D. (1998). *Neural networks: An Introductory Guide for Social Scientists*. Sage.

Genton, M. G. (2002). Classes of kernels for machine learning: a statistics perspective. *The Journal of Machine Learning Research*, 2:299–312.

Ghimire, B., Rogan, J., and Miller, J. (2010). Contextual land-cover classification: incorporating spatial dependence in land-cover classification models using random forests and the getis statistic. *Remote Sensing Letters*, 1(1):45–54.

Gill, B., Friebe, B., and Endo, T. (1991). Standard karyotype and nomenclature system for description of chromosome bands and structural aberrations in wheat (triticum aestivum). *Genome*, 34(5):830–839.

Glasbey, C. A. (1988). Normal distribution assumptions in discriminantion. In *Geoscience and Remote Sensing Symposium, 1988. IGARSS' 88. Remote Sensing: Moving Towards the 21st Century., International*, volume 3, pages 1789–1791.

Gualtieri, J. and Chettri, S. (2000). Support vector machines for classification of hyperspectral data. In *Geoscience and Remote Sensing Symposium, 2000. Proceedings. IGARSS 2000. IEEE 2000 International*, volume 2, pages 813–815. IEEE.

Gualtieri, J. and Comp, R. (1998). Support vector machines for hyperspectrtal remote sensing classification. In *27th AIPR Workshop: Advances in Computer Assisted Recognition*, pages 221–232, Washington, DC.

Guo, B., Damper, R. I., Gunn, S. R., and Nelson, J. D. (2008). A fast separability-based feature-selection method for high-dimensional remotely sensed image classification. *Pattern Recognition*, 41(5):1653–1662.

Gwet, K. (2002). Kappa statistic is not satisfactory for assessing the extent of agreement between raters. *Statistical Methods for Inter-Rater Reliability Assessment*, 1(6):1–6.

Gwet, K. L. (2014). *Handbook of Inter-Rater Reliability: The Definitive Guide to Measuring the Extent of Agreement among Raters*. Advanced Analytics, LLC.

Ham, J., Chen, Y., Crawford, M. M., and Ghosh, J. (2005). Investigation of the random forest framework for classification of hyperspectral data. *Geoscience and Remote Sensing, IEEE Transactions on*, 43(3):492–501.

Han, J.-w., Flemington, C., Houghton, A. B., Gu, Z., Zambetti, G. P., Lutz, R. J., Zhu, L., and Chittenden, T. (2001). Expression of bbc3, a pro-apoptotic bh3-only gene, is regulated by diverse cell death and survival signals. *Proceedings of the National Academy of Sciences*, 98(20):11318–11323.

Hand, D. J. (1997). *Construction and Assessment of Classification Rules*, volume 15. Wiley Chichester.

Hand, D. J. and Till, R. J. (2001). A simple generalisation of the area under the roc curve for multiple class classification problems. *Machine Learning*, 45(2):171–186.

Hanezar, B. and Dougherty, E. (2010). On the comparison of classifiers for microarray data. *Curr Bioinformatics*, 27:1675–1683.

Hanusz, Z. and Tarasińska, J. (2014). On multivariate normality tests using skewness and kurtosis. In *Colloquium Biometricum*, volume 44, pages 139–148.

Haykin, S. (1999). Multilayer perceptrons. *Neural Networks: A Comprehensive Foundation*, 2:156–255.

Hein, M. and Bousquet, O. (2005). Hilbertian metrics and positive definite kernels on probability measures. In *Proceedings of AISTATS*, volume 2005.

Hong, D. H. and Hwang, C. (2003). Support vector fuzzy regression machines. *Fuzzy Sets and Systems*, 138(2):271–281.

Hord, R. M. and Brooner, W. (1976). Land-use map accuracy criteria. *Photogrammetric Engineering and Remote Sensing*, 42(5).

Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2):251–257.

Howard, A. and Jebara, T. (2007). Learning monotonic transformations for classification. In *Advances in Neural Information Processing Systems*, pages 681–688.

Hoyle, M. (1973). Transformations: an introduction and a bibliography. *International Statistical Review/Revue Internationale de Statistique*, pages 203–223.

Hsu, C.-W. and Lin, C.-J. (2002). A comparison of methods for multiclass support vector machines. *Neural Networks, IEEE Transactions on*, 13(2):415–425.

Huang, C., Davis, L., and Townshend, J. (2002). An assessment of support vector machines for land cover classification. *International Journal of Remote Sensing*, 23:725–749.

Huang, C.-L. and Wang, C.-J. (2006). A ga-based feature selection and parameters optimizationfor support vector machines. *Expert Systems with applications*, 31(2):231–240.

Huang, R. and He, M. (2005). Band selection based on feature weighting for classification of hyperspectral data. *Geoscience and Remote Sensing Letters, IEEE*, 2(2):156–159.

Huang, X., Pan, W., Grindle, S., Han, X., Chen, Y., Park, S. J., Miller, L. W., and Hall, J. (2005). A comparative study of discriminating human heart failure etiology using gene expression profiles. *BMC bioinformatics*, 6(1–15):1.

Huberty, C. J. (1994). *Applied Discriminant Analysis*. Wiley New York.

Hughes, G. P. (1968). On the mean accuracy of statistical pattern recognizers. *Information Theory, IEEE Transactions on*, 14(1):55–63.

Hyvärinen, A. and Oja, E. (2000). Independent component analysis: algorithms and applications. *Neural Networks*, 13(4):411–430.

Jain, A. K. (1989). *Fundamentals of Digital Image Processing*. Prentice-Hall, Inc.

Jain, A. K., Duin, R. P., and Mao, J. (2000). Statistical pattern recognition: A review. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(1):4–37.

Janssen, L. L. and Vanderwel, F. J. (1994). Accuracy assessment of satellite derived land-cover data: a review. *Photogrammetric Engineering and Remote Sensing;(United States)*, 60(4).

Jensen, J. (2005). *Introductory Digital Image Processing: A Remote Sensing Perspective*. Prentice Hall, Upper Saddle River, NJ, Third edition.

Jimenez, L. O. and Landgrebe, D. A. (1998). Supervised classification in high-dimensional space: geometrical, statistical, and asymptotical properties of multivariate data. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 28(1):39–54.

Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. In *Machine Learning: ECML-98*, volume 1398 of *Lecture Notes in Computer Science*, pages 137–142. Springer Berlin Heidelberg.

Johnson, M., Wang, C., and Ramberg, J. (1979). Robustness of fisher's linear discriminant function to departures from normality. Technical report, Los Alamos Scientific Lab., NM (USA).

Johnson, N. L., Kotz, S., and Balakrishnan, N. (1994). 14: Lognormal distributions. *Continuous Univariate Distributions*, 1.

Johnson, R. and Wichern, D. (2007). *Applied Multivariate Statistical Analysis*. Pearson Education, Sixth edition.

Kaewpijit, S., Le Moigne, J., and El-Ghazawi, T. (2003). Automatic reduction of hyperspectral imagery using wavelet spectral analysis. *Geoscience and Remote Sensing, IEEE Transactions on*, 41(4):863–871.

Kalaichelvi, V. and Ali, A. S. (2012). Application of neural networks in character recognition. *International Journal of Computer Applications*, 52(12).

Kalkhan, M. A., Reich, R. M., and Czaplewski, R. L. (1995). Statistical properties of five indices in assessing the accuracy of remotely sensed data using simple random sampling. In *Proceedings ACSM/ASPRS Annual Convention and Exposition*, volume 2, pages 246–257.

Kanellopoulos, I., Wilkinson, G. G., Roli, F., and Austin, J. (2012). *Neurocomputation in Remote Sensing Data Analysis: Proceedings of Concerted Action COMPARES (Connectionist Methods for Pre-Processing and Analysis of Remote Sensing Data)*. Springer Science & Business Media.

Karhunen, J., Oja, E., Wang, L., Vigario, R., and Joutsensalo, J. (1997). A class of neural networks for independent component analysis. *Neural Networks, IEEE Transactions on*, 8(3):486–504.

Kavzoĝlu, T. (2001). *An Investigation of the design and use of feed-forward artificial neural networks in the classification of remotely sensed images*. PhD thesis, University of Nottingham.

Kavzoglu, T. and Kolkesen, I. (2009). A kernel function analysis for support vector machines for land cover classification. *International Journal of Applied Earth Observation and Geoinformation*, 11(5):530–541.

Keuchel, J., Naumann, S., Heiler, M., and Seigmund, A. (2003). Automatic landcover analysis for tenerife by by supervised classification using remotely sensed data. *Remote Sensing of Environment*, 86:530–541.

Khondoker, M., Dobson, R., Skirrow, C., and et al., A. S. (2013). A comparison of machine learning methods for classification using simulation with multiple real data examples from mental health studies. *Statistical Methods in Medical Research*, 0(0):1–20.

Kim, B. and Landgrebe, D. A. (1991). Hierarchical classifier design in high-dimensional numerous class cases. *IEEE Transactions on Geoscience and Remote Sensing*, 29(4):518–528.

Kittler, J. and Kml, L. (1978). Feature set search algorithms. *Pattern recognition and signal processing*, pages 41–60.

Knerr, S., Personnaz, L., and Dreyfus, G. (1990). Single-layer learning revisited: a stepwise procedure for building and training a neural network. In *Neurocomputing*, pages 41–50. Springer.

Kohonen, T. (1995). *Learning Vector Quantization*. Springer.

Komori, O. and Eguchi, S. (2015). Statistical and machine-learning methods for class prediction in high dimension. *Design and Analysis of Clinical Trials for Predictive Medicine*, 72:253.

Kon, M. A. and Plaskota, L. (2000). Information complexity of neural networks. *Neural Networks*, 13(3):365–375.

Kosorok, M. R., Ma, S., et al. (2007). Marginal asymptotics for the large p, small n paradigm: with applications to microarray data. *The Annals of Statistics*, 35(4):1456–1486.

Kotsiantis, S. B., Zaharakis, I., and Pintelas, P. (2007). Supervised machine learning: A review of classification techniques.

Koukoulas, S. and Blackburn, G. A. (2001). Introducing new indices for accuracy evaluation of classified images representing semi-natural woodland environments. *Photogrammetric Engineering and Remote Sensing*, 67(4):499–510.

Kriegel, H.-P., Kröger, P., Pryakhin, A., and Schubert, M. (2004). Using support vector machines for classifying large sets of multi-represented objects. In *SDM*, pages 102–113. SIAM.

Krogh, A., Vedelsby, J., et al. (1995). Neural network ensembles, cross validation, and active learning. *Advances in Neural Information Processing Systems*, 7:231–238.

Kruse, F., Lefkoff, A., Boardman, J., Heidebrecht, K., Shapiro, A., Barloon, P., and Goetz, A. (1993). The spectral image processing system (sips)interactive visualization and analysis of imaging spectrometer data. *Remote Sensing of Environment*, 44(2):145–163.

Kulkarni, A. and Kanal, L. N. (1976). An optimization approach to hierarchical classifier design. In *Proc. 3rd Int. Joint Conf. on Pattern Recognition, San Diego, CA*.

Kulkarni, V. Y. and Sinha, P. K. (2013). Random forest classifiers: a survey and future research directions. *International Journal of Advanced Computing*, 36(1):1144–1153.

Kulkarni, V. Y. and Sinha, P. K. (2014). Effective learning and classification using random forest algorithm. *International Journal of Engineering and Innovative Technology (IJEIT)*, 3:267–273.

Kullback, S. (1959). Information and statistics. *J. Wiley, New York*.

Kurzyński, M. W. (1983). The optimal strategy of a tree classifier. *Pattern Recognition*, 16(1):81–87.

Lachenbruch, P. (1975). *Discriminant Analysis*. Hafner Press, New York.

Lachenbruch, P. A. and Mickey, M. R. (1968). Estimation of error rates in discriminant analysis. *Technometrics*, 10(1):1–11.

Lachenbruch, P. A., Sneeringer, C., and Revo, L. (1973). Robustness of linear and quadratic discriminant functions to certain types of non normality. *Communications in statistics*, 1:39–56.

Landgrebe, D. A. (1980). The development of a spectral-spatial classifier for earth observational data. *Pattern Recognition*, 12(3):165–175.

Latifi, H., Nothdurft, A., and Koch, B. (2010). Non-parametric prediction and mapping of standing timber volume and biomass in a temperate forest: application of multiple optical/lidar-derived predictors. *Forestry*, 83(4):395–407.

Lawrence, R. L., Wood, S. D., and Sheley, R. L. (2006). Mapping invasive plants using hyperspectral imagery and breiman cutler classifications (randomforest). *Remote Sensing of Environment*, 100(3):356–362.

Lee, C. and Landgrebe, D. (1993). Feature extraction and classification algorithms for high dimensional data. Technical report, School of Electrical Engineering Purdue University.

Lee, J. W., Lee, J. B., Park, M., and Song, S. H. (2005). An extensive comparison of recent classification tools applied to microarray data. *Computational Statistics & Data Analysis*, 48(4):869–885.

Lee, T. and Richards, J. (1985). A low-cost classifier for multitemporal applications. *International Journal of Remote Sensing*, 6(8):1405–1417.

Lennon, M., Mercier, G., Mouchot, M., and Hubert-Moy, L. (2001). Independent component analysis as a tool for the dimensionality reduction and the representation of hyperspectral images. In *Geoscience and Remote Sensing Symposium, 2001. IGARSS'01. IEEE 2001 International*, volume 6, pages 2893–2895. IEEE.

Lerner, B., Guterman, H., Aladjem, M., et al. (1999). A comparative study of neural network based feature extraction paradigms. *Pattern Recognition Letters*, 20(1):7–14.

Leshno, M. and Spector, Y. (1996). Neural network prediction analysis: The bankruptcy case. *Neurocomputing*, 10(2):125–147.

Liaw, A. and Wiener, M. (2002). Classification and regression by randomforest. *R news*, 2(3):18–22.

Lippmann, R. P. (1987). An introduction to computing with neural nets. *ASSP Magazine, IEEE*, 4(2):4–22.

Lu, D., Mausel, P., Batistella, M., and Moran, E. (2004). Comparison of land cover classification methods in the Brazilian Amazon basin. *Photogrammetric Engineering and Remote Sensing*, 70:723–731.

Lu, D. and Weng, Q. (2007). An assessment of support vector machines for land cover classification. *International Journal of Remote Sensing*, 28(5):823–870.

Maes, F., Collignon, A., Vandermeulen, D., Marchal, G., and Suetens, P. (1997). Multimodality image registration by maximization of mutual information. *Medical Imaging, IEEE Transactions on*, 16(2):187–198.

Magnussen, S., Boudewyn, P., and Wulder, M. (2004). Contextual classification of landsat tm images to forest inventory cover types. *International Journal of Remote Sensing*, 25(12):2421–2440.

Man, M., Dyson, G., and et al., K. J. (2004). Evaluating methods for classifying expression data. *Biopharm Stat*, 14:1065–1084.

Mardia, K. (1970). Measures of multivariate skewness and kurtosis with applications. *Biometrika*, 57:519–530.

Mardia, K. (1974). Applications of some measures of multivariate skewness and kurtosis in testing normality and robustness studies. *Sankhya: The Indian Journal of Statistics*, B(36)(2):115–128.

Mardia, K. V. (1980). 9 tests of unvariate and multivariate normality. *Handbook of statistics*, 1:279–320.

Martinuzzi, S., Vierling, L. A., Gould, W. A., Falkowski, M. J., Evans, J. S., Hudak, A. T., and Vierling, K. T. (2009). Mapping snags and understory shrubs for a lidar-based assessment of wildlife habitat suitability. *Remote Sensing of Environment*, 113(12):2533–2546.

Mather, P. M. (2004). Computer processing of remotely sensed data: An introduction.

Mather, P. M. and Koch, M. (2011). *Computer Processing of Remotely-Sensed Images: An Introduction*. John Wiley & Sons.

Mathur, A. and Foody, G. (2008). Multiclass and binary svm classification: Implications for training and classification users. *Geoscience and Remote Sensing Letters, IEEE*, 5(2):241–245.

Matlab (2013a). *Matlab and Neural Network Toolbox release R2013a*. The MathWorks Inc., Natick, Massachusetts.

Matlab (2013b). *Matlab and Statistics Toolbox release R2013a*. The MathWorks Inc., Natick, Massachusetts.

McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4):115–133.

McLachlan, G. (2004). *Discriminant Analysis and Statistical Pattern Recognition*, volume 544. John Wiley & Sons.

Mecklin, C. J. and Mundfrom, D. J. (2004). An appraisal and bibliography of tests for multivariate normality. *International Statistical Review*, 72(1):123–138.

Meireles, M. R., Almeida, P. E., and Simões, M. G. (2003). A comprehensive review for industrial applicability of artificial neural networks. *Industrial Electronics, IEEE Transactions on*, 50(3):585–601.

Melgani, F. and Bruzzone, L. (2004). Classification of hyperspectral remote sensing images with support vector machines. *Geoscience and Remote Sensing, IEEE Transactions on*, 42(8):1778–1790.

Michelson, D., Liljeberg, B., and Pilesjo, P. (2000). Comparison of algorithms for classifying Swedish land cover using Landsat TM and ERS-I SAR data. *Remote Sensing of Environment*, 71:1–15.

Mountrakis, G., Im, J., and Ogole, C. (2011). An assessment of support vector machines for land cover classification. *ISPRS Journal of Photogrammatery and Remote Sensing*, 66:247–259.

Niemann, H. (1980). Linear and nonlinear mapping of patterns. *Pattern Recognition*, 12(2):83–87.

Olthof, I., King, D., and Lautenschlager, R. (2004). Mapping deciduous forest ice storm damage using landsat and environmental data. *Remote Sensing of Environment*, 89:484–496.

Oommen, T., Misra, D., Twarakavi, N. K., Prakash, A., Sahoo, B., and Bandopadhyay, S. (2008). An objective analysis of support vector machine based classification for remote sensing. *Mathematical Geosciences*, 40(4):409–424.

Opitz, D. and Maclin, R. (1999). Popular ensemble methods: An empirical study. *Journal of Artificial Intelligence Research*, pages 169–198.

Osborne, J. W. (2010). Improving your data transformations: Applying the box-cox transformation. *Practical Assessment, Research & Evaluation*, 15(12):1–9.

Otukei, J. and Blaschke, T. (2010). Land cover change assessment using decision trees, support vector machines and maximum likelihood classification algorithms. *International Journal of Applied Earth Observation and Geoinformation*, 12:S27–S31.

Pal, M. (2005). Random forest classifier for remote sensing classification. *International Journal of Remote Sensing*, 26(1):217–222.

Pal, M. (2008). Multiclass approaches for support vector machine based land cover classification. *arXiv preprint arXiv:0802.2411*.

Pal, M. and Mather, P. (2003). An assessment of the effectiveness of decision tree methods for land cover classification. *Remote Sensing of Environment*, 86:554–565.

Pal, M. and Mather, P. (2004). Assessment of the effectiveness of support vector machines for hyperspectral data. *Future Generation Computer Systems*, 20:1215–1225.

Paola, J. and Schowengerdt, R. (1995). A review and analysis of backpropagation neural networks for classification of remotely-sensed multi-spectral imagery. *International Journal of remote sensing*, 16(16):3033–3058.

Parsons, H. M., Ludwig, C., Günther, U. L., and Viant, M. R. (2007). Improved classification accuracy in 1-and 2-dimensional nmr metabolomics data using the variance stabilising generalised logarithm transformation. *BMC bioinformatics*, 8(1):234.

Pepe, M. S. (2003). *The Statistical Evaluation of Medical Tests for Classification and Prediction*. Oxford University Press, USA.

Pereira, F., Mitchell, T., and Botvinick, M. (2009). Machine learning classifiers and fmri: a tutorial overview. *Neuroimage*, 45(1):S199–S209.

Piper, S. (1983). The evaluation of the spatial accuracy of computer classification(for remote sensing). *Machine Processing of Remotely Sensed Data: Natural Resources Evaluation*, pages 303–310.

Pirooznia, M., Yang, J. Y., Yang, M. Q., and Deng, Y. (2008). A comparative study of different machine learning methods on microarray gene expression data. *BMC genomics*, 9(1):1.

Prasad, A. M., Iverson, L. R., and Liaw, A. (2006). Newer classification and regression tree techniques: bagging and random forests for ecological prediction. *Ecosystems*, 9(2):181–199.

Prinzie, A. and den Poel, D. V. (2008). Random forests for multiclass classification: Random multinomial logit. *Expert Systems with Applications*, 34:1721–1732.

Provost, F. and Fawcett, T. (1997). Analysis and visualization of classifier performance: Comparison under imprecise class and cost distributions. In *In Proceedings of the Third International Conference on Knowledge Discovery and Data Mining*.

Provost, F. and Fawcett, T. (2001). Robust classification for imprecise environments. *Machine Learning*, 42(3):203–231.

Provost, F. J., Fawcett, T., and Kohavi, R. (1998). The case against accuracy estimation for comparing induction algorithms. In *ICML*, volume 98, pages 445–453.

Pudil, P., Novovičová, J., and Kittler, J. (1994). Floating search methods in feature selection. *Pattern Recognition Letters*, 15(11):1119–1125.

Qiu, F. and Jensen, J. (2004). Opening the black box of neural networks for remote sensing image classification. *International Journal of Remote Sensing*, 25(9):1749–1768.

Quenouille, M. H. (2014). *Introductory Statistics*. Elsevier.

Quinlan, J. (1987). Generating production rules from decision trees. In *Proceedings of the 10th International Joint Conference on Artificial Intelligence*, volume 1, pages 304–307. Morgan Kaufmann Publishers Inc.

Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning*, 1(1):81–106.

Quinlan, J. R. et al. (1979). *Discovering Rules by Induction from Large Collections of Examples*. Expert systems in the micro electronic age. Edinburgh University Press.

Rais, S. (2015). *Spatio-temporal land cover changes and their imapcts on natural resource degradation in Karauli district, Rajasthan. A remote sensing and GIS based approach*. PhD thesis, Aligarh Muslim University.

Rao, C. R. (1982). Diversity and dissimilarity coefficients: a unified approach. *Theoretical Population Biology*, 21(1):24–43.

Rencher, A. C. (2003). *Methods of Multivariate Analysis*, volume 492. John Wiley & Sons.

Riani, M. and Atkinson, A. C. (2001). A unified approach to outliers, inuence, and transformations in discriminant analysis. *Journal of Computational and Graphical Statistics*, 10(3):513–544.

Richard, M. D. and Lippmann, R. P. (1991). Neural network classifiers estimate bayesian a posteriori probabilities. *Neural computation*, 3(4):461–483.

Richards, J. A. and Richards, J. A. (2008). *Remote Sensing Digital Image Analysis*. Springer, Fourth edition.

Rocke, D., Ideker, T., and et al., O. T. (2009). Papers on normalization, variable selection, classification or clustering of microarray data. *Bioinformatics*, 25:701–702.

Rosenfield, G. H. (1981). Analysis of variance of thematic mapping experiment data. *Photogrammetric Engineering and Remote Sensing*, 47(12):1685–1692.

Rosenfield, G. H. and Fitzpatrick-Lins, K. (1986). A coefficient of agreement as a measure of thematic classification accuracy. *Photogrammetric engineering and remote sensing*, 52(2):223–227.

Rosenfield, G. H., Fitzpatrick-Lins, K., and Ling, H. (1982). Sampling for thematic map accuracy testing. *Photogrammetric Engineering and Remote Sensing*, 48:131–137.

Rumelhart, D., Hinton, G., and Williams, R. (1986). Learning representations by back-propagating errors. *Nature*, 323:533–536.

Safavian, S. and Landgrebe, D. (1991). A survey of decision tree classifier methodology. *IEEE Transactions on Systems, Man, and Cybernetics*, 21(3):660–674.

Sahoo, G. et al. (2012). Analysis of parametric & non parametric classifiers for classification technique using WEKA. *International Journal of Information Technology and Computer Science (IJITCS)*, 4(7):43.

Sammon, J. W. (1969). A nonlinear mapping for data structure analysis. *IEEE Transactions on Computers*, (5):401–409.

Schapire, R. E. (1990). The strength of weak learnability. *Machine Learning*, 5(2):197–227.

Schapire, R. E. (2003). The boosting approach to machine learning: An overview. In *Nonlinear Estimation and Classification*, pages 149–171. Springer.

Schölkopf, B. and Burges, C. J. (1999). *Advances in Kernel Methods: Support Vector Learning*. MIT press.

Schölkopf, B., Smola, A., and Müller, K.-R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural computation*, 10(5):1299–1319.

Schölkopf, B., Sung, K.-K., Burges, C. J., Girosi, F., Niyogi, P., Poggio, T., and Vapnik, V. (1997). Comparing support vector machines with gaussian kernels to radial basis function classifiers. *IEEE Transactions on Signal Processing*, 45(11):2758–2765.

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, 34(1):1–47.

Seber, G. A. F. (2009). *Multivariate Observations*, volume 252. John Wiley & Sons.

Serpico, S. B., Moser, G., and Cattoni, A. F. (2007). Feature reduction for classification purpose. *Hyperspectral Data Exploitation: Theory and Applications*, pages 245–274.

Shao, Y. and S., L. R. (2012). Comparison of support vector machines, neural networks and cart algorithms for the land-cover classification using limited training data points. *ISPRS Journal of Photogrammetry and Remote Sensing*.

Shi, T. and Horvath, S. (2012). Unsupervised learning with random forest predictors. *Journal of Computational and Graphical Statistics*.

Smits, P., Dellepiane, S., and Schowengerdt, R. (1999). Quality assessment of image classification algorithms for land-cover mapping: a review and a proposal for a cost-based approach. *International journal of remote sensing*, 20(8):1461–1486.

Smits, P. C. (2002). Multiple classifier systems for supervised remote sensing image classification based on dynamic classifier selection. *IEEE Transactions on Geoscience and Remote Sensing*, 40(4):801–813.

South, S., Qi, J., and Lusch, D. P. (2004). Optimal classification methods for mapping agricultural tillage practices. *Remote Sensing of Environment*, 91(1):90–97.

Spackman, K. A. (1989). Signal detection theory: Valuable tools for evaluating inductive learning. In *Proceedings of the Sixth International Workshop on Machine Learning*, pages 160–163. Morgan Kaufmann Publishers Inc.

Srivastava, M. S. (2002). *Methods of Multivariate Statistics*. Wiley Interscience, New York.

Statnikov, A., Wang, L., and Aliferis, C. (2008). A comprehensive comparison of random forests and support vector machines for microarray-based cancer classification. *BMC Bioinformatics*, 9:319.

Stehman, S. V. (1992). Comparison of systematic and random sampling for estimating the accuracy of maps generated from remotely sensed data. *Photogrammetric Engineering and Remote Sensing*, 58(9):1343–1350.

Stehman, S. V. and Czaplewski, R. L. (1998). Design and analysis for thematic map accuracy assessment: fundamental principles. *Remote Sensing of Environment*, 64(3):331–344.

Strahler, A. H. (1980). The use of prior probabilities in maximum likelihood classification of remotely sensed data. *Remote Sensing of Environment*, 10(2):135–163.

Strickland, J. (2015). Classification trees using r. `https://www.linkedin.com/pulse/classification-trees-using-r-jeffrey-strickland-ph-d-cmsp-asep`. Accessed: 2016-02-24.

Su, H., Yang, H., Du, Q., and Sheng, Y. (2011). Semisupervised band clustering for dimensionality reduction of hyperspectral imagery. *Geoscience and Remote Sensing Letters, IEEE*, 8(6):1135–1139.

Swain, P. H. and Hauska, H. (1977). The decision tree classifier: Design and potential. *Geoscience Electronics, IEEE Transactions on*, 15(3):142–147.

Szuster, B. W., Chen, Q., and Borger, M. (2011). A comparison of classification techniques to support land and land cover analysis in tropical coastal zones. *Applied Geography*, 31:525–532.

Tan, A. C. and Gilbert, D. (2003). An empirical comparison of supervised machine learning techniques in bioinformatics. In *Proceedings of the First Asia-Pacific bioinformatics conference on Bioinformatics 2003-Volume 19*, pages 219–222. Australian Computer Society, Inc.

Team, R. C. (2013). *R: A Language and Environment for Statistical Computing*. Vienna, Austria.

Thenkabail, P. S. (2015). *Remotely Sensed Data Characterization, Classification and Accuracies*. CRC Press.

Tsai, C.-F. and Wu, J.-W. (2008). Using neural network ensembles for bankruptcy prediction and credit scoring. *Expert Systems with Applications*, 34(4):2639–2649.

Tso, B. and Mather, P. M. (2009). *Classification Methods for Remotely Sensed Data*. CRC Press, Second edition.

Tso, B. and Olsen, R. C. (2005). Combining spectral and spatial information into hidden markov models for unsupervised image classification. *International Journal of Remote Sensing*, 26(10):2113–2133.

Tumer, K. and Ghosh, J. (1995). Theoretical foundations of linear and order statistics combiners for neural pattern classifiers. *IEEE Trans. Neural Networks*.

Ujiie, H., Omachi, S., and Aso, H. (2002). A discriminant function considering normality improvement of the distribution. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, volume 2, pages 224–227. IEEE.

Van der Meer, F. D. and De Jong, S. M. (2011). *Imaging Spectrometry: Basic Principles and Prospective Applications*, volume 4. Springer Science & Business Media.

Van Genderen, J. and Lock, B. (1977). Testing land-use map accuracy. *Photogrammetric engineering and remote sensing*, 43(9).

Vapnik, V. (1979). *Estimation of Dependences based on Empirical Data [in Russian]*. English translation: Springer-Verlag New York.

Vapnik, V. N. and Vapnik, V. (1998). *Statistical Learning Theory*, volume 1. Wiley New York.

Veropoulos, K., Cristianini, N., and Campbell, C. (1999). The application of support vector machines to medical decision support: a case study. *Advanced Course in Artificial Intelligence*, pages 1–6.

Vieira, C. and Mather, P. (2001). On the assessment of the spatial reliability of thematic images. ed. peter j. halls. *Innovations in GIS 8: Spatial Information and the Environment*, pages 88–101.

Vladimir, V. N. and Vapnik, V. (1995). *The nature of statistical learning theory*. Springer Verlag, New York.

Wakabayashi, T., Tsuruoka, S., Kimura, F., and Miyake, Y. (1993). On the size and variable transformation of feature vector for handwritten character recognition. *Trans. IEICE Japan J76-D-II (12)*, pages 2495–2503.

Wang, G., Gertner, G., and Anderson, A. (2005). Sampling design and uncertainty based on spatial variability of spectral variables for mapping vegetation cover. *International Journal of Remote Sensing*, 26(15):3255–3274.

Waske, B. and Braun, M. (2009). Classifier ensembles for land cover mapping using multitemporal sar imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 64(5):450–457.

Wilkinson, G., Fierens, F., and Kanellopoulos, I. (1995). Integration of neural and statistical approaches in spatial data classification. *Geographical Systems*, 2(1):1–20.

Witten, I. H. and Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann.

Wolpert, D. H. (1992). Stacked generalization. *Neural networks*, 5(2):241–259.

Wongpakaran, N., Wongpakaran, T., Wedding, D., and Gwet, K. L. (2013). A comparison of cohen's kappa and gwet's ac1 when calculating inter-rater reliability coefficients: a study conducted with personality disorder samples. *BMC Medical Research Methodology*, 13(1):1.

Woodcock, C. E., Strahler, A. H., and Jupp, D. L. (1988a). The use of variograms in remote sensing: Ii. real digital images. *Remote Sensing of Environment*, 25(3):349–379.

Woodcock, C. E., Strahler, A. H., and Jupp, D. L. (1988b). The use of variograms in remote sensing: I.scene models and simulated images. *Remote Sensing of Environment*, 25(3):323–348.

Yeo, I.-K. and Johnson, R. A. (2000). A new family of power transformations to improve normality or symmetry. *Biometrika*, 87(4):954–959.

Yousefi, M., Hua, J., and Dougherty, E. (2011a). Multiple rule bias in the comparison of classification rules. *Bioinformatics*, 27:1675–1683.

Yousefi, M. R., Hua, J., and Dougherty, E. R. (2011b). Multiple-rule bias in the comparison of classification rules. *Bioinformatics*, 27(12):1675–1683.

Zadkarami, M. R. and Rowhani, M. (2010). Application of skew normal in classification of satellite images. *Journal of Data Science*, 8(4):597–606.

Zakariah, M. (2014). Classification of large datasets using random forest algorithm in various applications: Survey. *International Journal of Engineering and Innovative Technology*, 4(3).

Zhang, D. and Jain, A. K. (2006). Advances in biometrics. In *International Conference, ICB 2006, Hong Kong, China, January 5-7, 2006, Proceedings*, volume 3832.

Zhang, G., Hu, M. Y., Patuwo, B. E., and Indro, D. C. (1999). Artificial neural networks in bankruptcy prediction: General framework and cross-validation analysis. *European Journal of Operational Research*, 116(1):16–32.

Zhang, G. P. (2000). Neural networks for classification: a survey. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 30(4):451–462.

Zhuang, X., Engel, B., Xiong, X., and Johannsen, C. (1995). Analysis of classification results of remotely sensed data and evaluation of classification algorithms. *Photogrammetric Engineering and Remote Sensing*, 61:427–433.